

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Explorando LLMs Abertos para Classificação de Discurso de Ódio em Jogos Online

Victor Rozzatti Tornisiello

Monografia - MBA em Inteligência Artificial e Big Data

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Victor Rozzatti Tornisiello

Explorando LLMs Abertos para Classificação de Discurso de Ódio em Jogos Online

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Ricardo Marcondes Marcacini

Versão original

São Carlos

2024

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTA TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO PARA FINS DE ESTUDO E
PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi, ICMC/USP, com os dados
fornecidos pelo(a) autor(a)

S856m	Tornisiello, Victor Rozzatti Explorando LLMs Abertos para Classificação de Discurso de Ódio em Jogos Online / Victor Rozzatti Tornisiello ; orientador Ricardo Marcondes Marcacini. – São Carlos, 2024. 60 p. : il. (algumas color.) ; 30 cm. Monografia (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2024. 1. LaTeX. 2. abnTeX. 3. Classe USPSC. 4. Editoração de texto. 5. Normalização da documentação. 6. Tese. 7. Dissertação. 8. Documentos (elaboração). 9. Documentos eletrônicos. I. Marcacini, Ricardo Marcondes, orient. II. Título.
-------	--

Victor Rozzatti Tornisiello

Exploring Open LLMs for Hate Speech Classification in Online Games

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Prof. Dr. Ricardo Marcondes Marcacini

Original version

São Carlos

2024

AGRADECIMENTOS

Aos meus pais, Raquel e Marcos, pelo incentivo, confiança e apoio sempre presentes durante toda minha vida.

À Universidade de São Paulo e ao Instituto de Ciências Matemáticas e de Computação e seus colaboradores, pela estrutura e ensino fornecidos durante o curso.

Ao Prof. Dr. Ricardo Marcondes Marcacini pela orientação e apoio durante o desenvolvimento deste projeto, desde a formulação de sua proposta até a redação da presente monografia.

RESUMO

TORNISIELLO, V. R. **Explorando LLMs Abertos para Classificação de Discurso de Ódio em Jogos Online**. 2024. 60 p. Monografia (MBA em Inteligência Artificial e Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2024.

O discurso de ódio é marcadamente presente em comunidades de jogos online. Motivado por isso, este projeto tem como objetivo o estudo da viabilidade da utilização de grandes modelos de linguagem abertos e com quantidades reduzidas de parâmetros na identificação e avaliação de linguagem ofensiva e discurso de ódio implícito em texto. Para atingir este objetivo, tendo como foco jogos online, foi proposto um pipeline para a classificação e avaliação de linguagem ofensiva e discurso de ódio, e cada uma de suas etapas foi avaliada de forma quantitativa ou qualitativa através de experimentos realizados utilizando o conjunto de dados ToxiGen, o modelo Llama3, de 8 bilhões de parâmetros, e os modelos Mistral e OpenChat, de 7 bilhões de parâmetros cada. Os resultados mostram que o uso de grandes modelos de linguagem na classificação binária de discurso de ódio implícito é viável, com os modelos escolhidos superando modelos de arquiteturas mais simples especializados nesse tipo de tarefa. Adicionalmente, o uso de grandes modelos de linguagem mostrou-se viável também na execução de tarefas mais complexas, como na identificação dos grupos minoritários alvo mencionados, na avaliação do grau de toxicidade e na elaboração de explicações sobre os textos.

Palavras-chave: Discurso de ódio. Linguagem Ofensiva. Grandes Modelos de Linguagem. Classificação.

ABSTRACT

TORNISIELLO, V. R. **Exploring Open LLMs for Hate Speech Classification in Online Games**. 2024. 60 p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2024.

Hate speech is markedly present in online gaming communities. Motivated by this, this project aims to study the viability of using open large language models with a reduced number of parameters in identifying and assessing offensive language and implicit hate speech in text. To achieve this goal, focusing on online games, a pipeline for offensive language and hate speech classification and assessment was proposed, and each of its stages was evaluated in a quantitative or qualitative way through experiments carried out using the ToxiGen dataset, the Llama3 model, with 8 billion parameters, and the Mistral and OpenChat models, with 7 billion parameters each. The results show that the use of large language models in implicit hate speech binary classification is viable, with the chosen models outperforming simpler architectures models that are specialized in this task. Furthermore, the use of large language models has also shown to be viable on more complex tasks, such as identifying the mentioned target minority group, assessing the toxicity level and explaining the texts.

Keywords: Hate speech. Offensive Language. Large Language Models. Classification.

LISTA DE FIGURAS

Figura 1 – Geração de texto autorregressiva	22
Figura 2 – Exemplo de <i>prompt zero-shot</i> para classificação de texto	24
Figura 3 – Exemplo de <i>prompt zero-shot</i> para resumo de texto	24
Figura 4 – Exemplo de <i>prompt zero-shot</i> para responder perguntas	25
Figura 5 – Exemplo de <i>prompt few-shot</i> para classificação de texto	26
Figura 6 – Exemplo de <i>prompt few-shot</i> para resumo de texto	27
Figura 7 – Exemplo de <i>prompt few-shot</i> para responder perguntas	28
Figura 8 – Sistema de RAG	28
Figura 9 – Diagrama de processos	32
Figura 10 – Quantidade de amostras tóxicas e não tóxicas no subconjunto de treinamento do ToxiGen-HumanVal	34
Figura 11 – Quantidade de amostras tóxicas e não tóxicas no subconjunto de teste do ToxiGen-HumanVal	34
Figura 12 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal	35
Figura 13 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal após o pré-processamento	36
Figura 14 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal	36
Figura 15 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal após o pré-processamento	37
Figura 16 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal após a agregação de categorias	37
Figura 17 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal após a agregação de categorias	38
Figura 18 – <i>Prompt</i> de sistema <i>zero-shot</i>	39
Figura 19 – <i>Prompt</i> de usuário <i>zero-shot</i>	39
Figura 20 – RAG para <i>prompt few-shot</i>	39
Figura 21 – <i>Prompt</i> de sistema <i>few-shot</i> base	40
Figura 22 – <i>Prompt</i> de sistema <i>few-shot</i> com exemplos	41
Figura 23 – <i>Prompt</i> de sistema <i>zero-shot</i> para classificação do grupo alvo	42
Figura 24 – <i>Prompt</i> de sistema <i>few-shot</i> para classificação do grupo alvo	43
Figura 25 – <i>Prompt</i> de sistema <i>zero-shot</i> para definição do nível de toxicidade	44
Figura 26 – <i>Prompt</i> de usuário para definição do nível de toxicidade	44
Figura 27 – <i>Prompt</i> de sistema <i>few-shot</i> para definição do nível de toxicidade	45
Figura 28 – <i>Prompt</i> de sistema para geração de explicações	46

Figura 29 – <i>Prompt</i> de usuário para geração de explicações	46
Figura 30 – Matriz de confusão obtida utilizando o modelo HateBERT	48
Figura 31 – Matriz de confusão obtida utilizando o modelo RoBERTa	48
Figura 32 – Matriz de confusão da classificação binária utilizando o modelo Open-Chat e <i>prompt few-shot</i>	49
Figura 33 – Matriz de confusão da classificação multiclasse utilizando o modelo Llama3 e <i>prompt few-shot</i>	51
Figura 34 – Matriz de confusão da classificação multiclasse agregada utilizando o modelo Llama3 e <i>prompt few-shot</i>	53
Figura 35 – Quantidade de amostras por categoria de nível de toxicidade	54
Figura 36 – Matriz de confusão da classificação dos níveis de toxicidade utilizando o modelo OpenChat e <i>prompt zero-shot</i>	55

LISTA DE TABELAS

Tabela 1	– Métricas obtidas com o HateBERT e RoBERTa na classificação binária	47
Tabela 2	– Métricas obtidas com os LLMs na classificação binária	49
Tabela 3	– Erros cometidos pelo modelo OpenChat na classificação binária utilizando <i>prompt few-shot</i>	50
Tabela 4	– Métricas obtidas com os LLMs na classificação do grupo alvo	51
Tabela 5	– Confusões entre grupos alvo similares	52
Tabela 6	– Métricas obtidas com os LLMs na classificação dos grupos alvo agregados	52
Tabela 7	– Exemplos de amostras rotuladas incorretamente	53
Tabela 8	– Métricas obtidas com os LLMs na classificação do nível de toxicidade .	55
Tabela 9	– Explicações geradas para um texto tóxico	56
Tabela 10	– Explicações geradas para um texto não tóxico	56

SUMÁRIO

1	INTRODUÇÃO	19
1.1	Hipóteses e Objetivos	19
1.2	Organização do Texto	20
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	Large Language Models (LLMs)	21
2.1.1	In-Context Learning	22
2.1.2	Instruction Tuning	22
2.2	Engenharia de Prompt	23
2.2.1	Zero-shot Learning	23
2.2.1.1	Classificação de Texto	23
2.2.1.2	Sumarização de Texto	24
2.2.1.3	Perguntas e Respostas	24
2.2.2	Few-shot Learning	25
2.2.2.1	Classificação de Texto	25
2.2.2.2	Sumarização de Texto	26
2.2.2.3	Perguntas e Respostas	26
2.3	Retrieval Augmented Generation (RAG)	26
2.4	Trabalhos Relacionados	29
3	DESENVOLVIMENTO	31
3.1	Objetivo e Solução Proposta	31
3.2	Conjunto de Dados	32
3.2.1	Pré-processamento	33
3.3	Metodologia	36
3.3.1	Dados de Entrada	37
3.3.2	Classificação Binária	38
3.3.2.1	Classificação Binária Zero-Shot	38
3.3.2.2	Classificação Binária Few-Shot	38
3.3.3	Classificação Multiclasse	40
3.3.3.1	Classificação Multiclasse Zero-Shot	40
3.3.3.2	Classificação Multiclasse Few-Shot	40
3.3.4	Nível de Toxicidade	41
3.3.4.1	Geração Zero-Shot	42
3.3.4.2	Geração Few-Shot	42
3.3.5	Explicação	43

3.4	Modelos Utilizados	46
4	AVALIAÇÃO EXPERIMENTAL	47
4.1	Classificação Binária	47
4.2	Classificação Multiclasse	50
4.3	Nível de Toxicidade	53
4.4	Explicações	55
5	CONCLUSÕES	57
5.1	Limitações	57
5.2	Melhorias e Trabalhos Futuros	58
	REFERÊNCIAS	59

1 INTRODUÇÃO

Durante as últimas décadas, com a evolução dos computadores e das redes de computadores, o mundo passou por grandes transformações na forma como está conectado e em como as pessoas se comunicam, até chegar ao estado em que se encontra atualmente. Uma das muitas atividades possibilitadas por essas transformações foi a criação e popularização de uma grande variedade de jogos online. Aproximadamente 3,09 bilhões de pessoas no mundo todo jogaram jogos online ativamente durante o ano de 2023 e a tendência é que esse número continue crescendo nos próximos anos (Howarth, 2024).

Uma das particularidades dos jogos online é a possibilidade de interação entre os jogadores, que podem se comunicar através de texto ou voz, em tempo real. Essa característica tem muitos pontos positivos: ela possibilita a formação de novas amizades e ajuda a desenvolver habilidades de comunicação e cooperação em equipe. É comum até mesmo o aprendizado, em algum grau, de novos idiomas, dependendo do servidor em que o jogo está hospedado ou do idioma praticado pela maioria da base de jogadores. Porém, ela traz consigo também um ponto muito negativo: a possibilidade da utilização de linguagem ofensiva e disseminação de discurso de ódio direcionado a outros jogadores. Em um estudo conduzido sobre jogadores dos Estados Unidos, constatou-se que 66% dos jogadores que possuem entre 13 a 17 anos sofreram algum tipo de assédio durante suas sessões de jogos online. Entre os adultos (18 a 45 anos), esse número sobe para 80% (ADL, 2022).

As estratégias de moderação empregadas pelas desenvolvedoras dos jogos se baseiam principalmente em mecanismos reativos que envolvem ações do usuário, como o desligamento das funções de chat de texto e/ou de voz e a denúncia de comportamento inadequado de outros jogadores durante ou ao final das partidas. Também existem estratégias de moderação automatizada, que detectam e filtram o uso de linguagem ofensiva explícita baseando-se em palavras-chave ou através da utilização de técnicas de aprendizado de máquina (Ng; Lim; Yoder, 2024), mas que podem ser burladas com o uso de variantes das palavras ou com o uso de expressões implícitas.

Portanto, a utilização de métodos mais sofisticados e proativos de moderação em jogos online se faz necessária para assegurar a melhor experiência de jogo possível para sua base de jogadores, livre de discurso de ódio e linguagem ofensiva.

1.1 Hipóteses e Objetivos

Dados os avanços nos estudos no campo da inteligência artificial nas últimas décadas e, em especial, ao surgimento dos Large Language Models (LLMs) nos últimos anos, é razoável pensar que estratégias mais sofisticadas para a moderação da comunicação entre

jogadores em jogos online possam ser adotadas, identificando e avaliando com maior assertividade o uso de linguagem ofensiva e discurso de ódio em sentenças implícitas.

Assim, o objetivo deste projeto é estudar a viabilidade da utilização de LLMs abertos e com quantidades reduzidas de parâmetros na identificação e avaliação de linguagem ofensiva e discurso de ódio implícito em texto, permitindo seu uso em ambientes com recursos computacionais de baixo custo, ao mesmo tempo em que preserva o desempenho nas tarefas executadas.

Nessas condições, o *fine-tuning* do modelo é facilitado, oferecendo maior flexibilidade para sua utilização na resolução de problemas ainda mais específicos. Além disso, a possibilidade de realizar inferência em máquinas comuns elimina a necessidade da utilização de modelos com centenas de bilhões de parâmetros através de APIs pagas, reduzindo consideravelmente os custos de operação e garantindo maior privacidade dos dados. Por fim, um modelo de tamanho reduzido também implica em um tempo de inferência menor, o que permite o desenvolvimento de aplicações mais rápidas.

1.2 Organização do Texto

No Capítulo 2 é apresentada a fundamentação teórica do projeto, com uma revisão da terminologia e conceitos utilizados em seu desenvolvimento e a apresentação e revisão de trabalhos relacionados. Nos capítulos 3 e 4, são apresentadas a metodologia adotada e a avaliação experimental, respectivamente. Finalmente, no capítulo 5, são apresentadas a conclusão do trabalho, limitações e pontos de melhoria futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, serão apresentados e discutidos conceitos fundamentais para o desenvolvimento deste projeto, bem como trabalhos de pesquisa relacionados ao tema de classificação de discurso de ódio utilizando modelos de linguagem.

2.1 Large Language Models (LLMs)

Large Language Models são grandes modelos de linguagem baseados na arquitetura Transformer (Vaswani *et al.*, 2023) e utilizados no Processamento de Linguagem Natural (PLN). Eles possuem desde poucos bilhões até centenas de bilhões de parâmetros, são treinados em quantidades massivas de dados e superam seus antecessores significativamente na realização de diversas tarefas de PLN, especialmente na geração de texto.

Em comparação com outros modelos de linguagem menores, os LLMs possuem habilidades chave para a execução de tarefas complexas, das quais destacam-se a capacidade de aprender em contexto e a capacidade de seguir instruções (Zhao *et al.*, 2023). Estas habilidades serão discutidas em mais detalhes nas seções 2.1.1 e 2.1.2.

Os LLMs revolucionaram a forma como os seres humanos interagem com os modelos de linguagem, tanto durante o seu desenvolvimento quanto durante sua utilização. Diferente do que ocorre com modelos de linguagem menores, a principal forma de interação com LLMs é através de interfaces de *prompt*. Um *prompt* refere-se à entrada de texto fornecida ao modelo, como uma tarefa a ser executada, por exemplo. Os seres humanos precisam entender como os LLMs funcionam e estruturar as tarefas de acordo, para que os LLMs possam compreendê-las e executá-las (Zhao *et al.*, 2023).

De modo geral, os LLMs mais recentes funcionam prevendo a próxima palavra a partir de uma sequência de palavras fornecida como entrada e sempre geram uma palavra por vez como saída. Assim, para gerar texto, é necessário realizar uma série de etapas de inferência, incrementando as entradas do modelo em cada uma delas com a saída gerada na etapa anterior, até que uma condição de parada seja atingida, como ilustrado na figura 1. Este processo é chamado de geração autorregressiva e os LLMs são também chamados de modelos autorregressivos.

Dado o seu desempenho excepcional na geração de texto, os LLMs têm se popularizado bastante nos últimos anos e vêm sendo utilizados em diversas aplicações, como no chatbot ChatGPT¹.

¹ <https://openai.com/chatgpt>

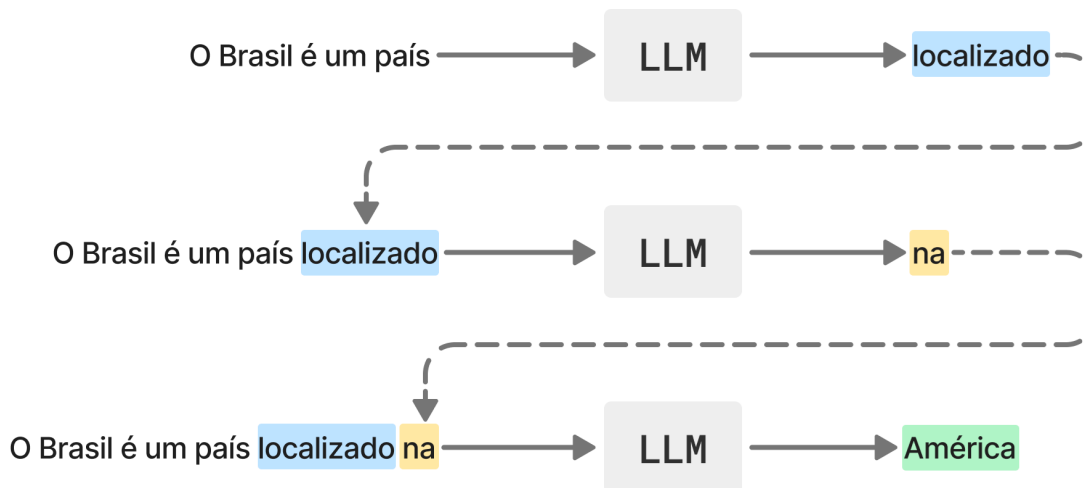


Figura 1 – Geração de texto autorregressiva

2.1.1 In-Context Learning

Dada uma instrução fornecida ao modelo, o *in-context learning*, ou aprendizado em contexto, é a habilidade de gerar as saídas desejadas apenas completando a sequência de palavras do texto de entrada – o contexto – sem a necessidade de treinamento adicional ou *fine-tuning*. Essa habilidade foi formalmente introduzida pelo GPT-3, que possui 175 bilhões de parâmetros². Vale ressaltar que a viabilidade do uso de aprendizado em contexto para a realização de tarefas de PLN depende do modelo utilizado e da tarefa específica a ser realizada (Zhao *et al.*, 2023).

2.1.2 Instruction Tuning

Instruction-tuning é o processo de *fine-tuning* de LLMs em conjuntos de dados contendo instruções específicas em linguagem natural para a realização de determinadas tarefas. Este processo, além de otimizar o desempenho do modelo na execução das tarefas presentes no conjunto de dados, também aumenta a habilidade do modelo em seguir instruções no geral, aprimorando sua capacidade de generalização para a realização de outras novas tarefas (Aw *et al.*, 2023).

Em comparação com o pré-treinamento dos modelos base de LLMs, o *instruction-tuning* geralmente consome muito menos recursos computacionais para ser realizado e permite que os LLMs atinjam performances próximas às de seres humanos na realização de muitas tarefas (Aw *et al.*, 2023).

Além de aprimorar a habilidade dos LLMs em seguir instruções humanas, o *instruction-tuning* também aprimora a habilidade dos LLMs em fornecer saídas mais próximas daquelas presentes no conjunto de dados utilizado para o *fine-tuning*, geralmente escritas por humanos. Isso permite que os modelos produzam saídas mais controláveis,

² <https://aws.amazon.com/what-is/large-language-model>

previsíveis e semelhantes à escrita humana, o que facilita o uso de métricas na avaliação do modelo e o seu alinhamento à ética humana. Consequentemente, o *instruction-tuning* produz LLMs mais próximos aos humanos, tanto em capacidade quanto na qualidade dos textos gerados (Aw *et al.*, 2023).

2.2 Engenharia de Prompt

Devido à grande quantidade de parâmetros dos modelos e à enorme quantidade de dados necessária para um treinamento, treinar um LLM é normalmente uma tarefa inviável, pois exige muitos recursos de memória, computação e tempo. Porém, os LLMs são muito flexíveis, e um mesmo modelo ajustado para seguir instruções pode ser utilizado para completar uma variedade de tarefas de PLN sem a necessidade de treinamento adicional ou *fine-tuning*, através de técnicas de engenharia de *prompt*.

No contexto de LLMs, a engenharia de *prompt* é o processo de otimização dos *prompts* fornecidos ao modelo para a obtenção das saídas desejadas. Quaisquer mudanças no *prompt* podem gerar saídas significativamente diferentes, o que torna este um processo iterativo e experimental e que deu origem a uma variedade de técnicas diferentes. Dentre elas, serão utilizadas neste projeto as técnicas de *zero-shot learning* e *few-shot learning*.

2.2.1 Zero-shot Learning

No âmbito de LLMs, a técnica de *zero-shot learning* é caracterizada pela utilização de um *prompt* diferente daquele utilizado durante o treinamento do modelo, sem fornecer nenhum exemplo de saída desejada. Geralmente, é fornecida ao modelo apenas uma instrução acompanhada de algum contexto, o que é suficiente para muitos casos de uso.

Nas subseções seguintes, são apresentados exemplos de *zero-shot learning* aplicados às tarefas de classificação de texto, sumarização de texto e de responder perguntas.

2.2.1.1 Classificação de Texto

Seguindo a temática deste projeto, pode-se tomar como exemplo a tarefa de classificação de texto entre duas categorias distintas: neutro ou tóxico. Para classificar o texto “*They have been bred to be good at sports and entertainment, but not much else.*”, retirado de (Hartvigsen *et al.*, 2022), utilizando *zero-shot learning*, o *prompt* apresentado na figura 2 poderia ser utilizado.

Nota-se que, além de fornecer a instrução ao modelo, também destacou-se o texto cujo conteúdo deve ser analisado e foi fornecido também o início da resposta no formato desejado, para induzir o modelo a responder com o nome da classe. A partir daí, o modelo deve gerar a próxima palavra da sequência, que esperamos ser *neutral* ou *toxic*.

```
Classify the below text between the following categories: neutral, toxic

Text: They have been bred to be good at sports and entertainment, but not much
else.

Category:
```

Figura 2 – Exemplo de *prompt zero-shot* para classificação de texto

2.2.1.2 Sumarização de Texto

A tarefa de sumarização de texto nada mais é do que uma tarefa de geração de texto, o que torna o uso de LLMs particularmente adequado para realizá-la. A saída do modelo na sumarização é especialmente influenciada pela entrada fornecida, já que, além da instrução, é necessário fornecer também o texto original a ser resumido. Na figura 3 está ilustrado um exemplo de *prompt* adequado para esta tarefa, utilizando um trecho do relatório “*Hate is No Game: Hate and Harassment in Online Games 2022*”, produzido pela *Anti-Defamation League* (ADL, 2022).

Novamente, a instrução foi fornecida ao modelo, seguida do texto cujo resumo deve ser realizado e, por fim, do início da resposta no formato como a desejamos, induzindo o modelo a gerar o resumo.

```
Write a summary of the below text:

Text: For the fourth consecutive year, harassment in online games has not
decreased. More than four out of five adult gamers experience harassment in
online multiplayer games—over 67 million American adults. Among adult online
multiplayer gamers, severe harassment increased from 71% to 77%, with roughly
60 million American adults reporting experiences including physical threats,
stalking, and sustained harassment.

Summary:
```

Figura 3 – Exemplo de *prompt zero-shot* para resumo de texto

2.2.1.3 Perguntas e Respostas

A tarefa de responder a perguntas também é um caso particular de geração de texto, tornando os LLMs os modelos mais apropriados para realizá-la. Podemos estruturar a entrada fornecida ao modelo em seções lógicas contendo a instrução, um contexto opcional,

a pergunta a ser respondida e, ao final, utilizar uma frase como “*Answer:*” ou “Resposta:”, para induzir o modelo a gerar a resposta correspondente. Na figura 4 está ilustrado um exemplo de prompt adequado, utilizando um trecho do relatório em (ADL, 2022).

Answer the question given the context

Context: For the fourth consecutive year, harassment in online games has not decreased. More than four out of five adult gamers experience harassment in online multiplayer games—over 67 million American adults. Among adult online multiplayer gamers, severe harassment increased from 71% to 77%, with roughly 60 million American adults reporting experiences including physical threats, stalking, and sustained harassment.

Question: What was the increase on severe harassment among adult players in multiplayer games?

Answer:

Figura 4 – Exemplo de *prompt zero-shot* para responder perguntas

2.2.2 Few-shot Learning

A técnica de *few-shot learning* é também caracterizada pela utilização de um *prompt* distinto daquele empregado durante o treinamento do modelo, porém, contendo pelo menos um exemplo de saída desejada. Os exemplos fornecidos no *prompt* fornecem mais contexto da tarefa a ser executada ao modelo, condicionando-o a gerar saídas mais adequadas e próximas ao formato desejado, como demonstrado em (Brown *et al.*, 2020).

Apesar dos *prompts zero-shot* funcionarem bem para muitas aplicações, existem tarefas mais complexas e que exigem saídas melhores definidas por parte dos modelos, de forma que apenas fornecer uma instrução e algum contexto não seja o suficiente para executá-las de forma satisfatória. Nesses casos, o *few-shot learning* pode ser adequado. O número de exemplos fornecidos é arbitrário e a elaboração de um *prompt* com exemplos adequados é um processo iterativo e experimental.

Nas próximas subseções, são apresentados exemplos de *few-shot learning* aplicados às tarefas de classificação de texto, sumarização de texto e de responder perguntas.

2.2.2.1 Classificação de Texto

Para classificar o texto “*You can tell if a woman is a good person or not by how much she covers her skin*”, retirado de (Hartvigsen *et al.*, 2022), entre as categorias *neutral* e *toxic*, utilizando *few-shot learning*, pode ser utilizado o *prompt* apresentado na figura 5.

Observa-se que, além da instrução principal e antes de fornecer o texto a ser classificado, foi fornecido também um exemplo de tarefa completada, com a saída no formato desejado, e foi solicitado ao modelo que utilize-o para auxiliar na elaboração de sua resposta. Além disso, foram utilizados separadores, “====”, para separar o exemplo do restante do *prompt* e destacá-lo.

Classify the below text between the following categories: neutral, toxic. Use the example to help you respond.

Example:

====

Text: They have been bred to be good at sports and entertainment, but not much else.

Category: toxic

====

Text: You can tell if a woman is a good person or not by how much she covers her skin

Category:

Figura 5 – Exemplo de *prompt few-shot* para classificação de texto

2.2.2.2 Sumarização de Texto

Analogamente, para a tarefa de sumarização utilizando *few-shot learning*, pode-se fornecer como exemplo um texto seguido de seu resumo, separados do texto de interesse por um separador, como no *prompt* da figura 6. Os textos utilizados como exemplo foram retirados do relatório em (ADL, 2022).

2.2.2.3 Perguntas e Respostas

Para tarefas envolvendo perguntas e respostas podemos manter a estrutura utilizada anteriormente no *prompt zero-shot*, com a adição de um exemplo de resposta antes da pergunta de interesse e separado por um separador, como ilustrado na figura 7. Os textos utilizados como exemplo também foram retirados do relatório em (ADL, 2022).

2.3 Retrieval Augmented Generation (RAG)

Apesar de excelentes na geração de texto, os LLMs ainda são modelos de redes neurais e possuem limitações. Textos gerados por LLMs são baseados em padrões aprendidos durante o treinamento, onde podem existir informações desatualizadas ou incorretas. Essas informações são incorporadas pelo modelo e certamente afetam a qualidade de suas

Write a summary of the below text. Use the example to help you respond.

Example:

=====

Text: For the fourth consecutive year, harassment in online games has not decreased. More than four out of five adult gamers experience harassment in online multiplayer games—over 67 million American adults. Among adult online multiplayer gamers, severe harassment increased from 71% to 77%, with roughly 60 million American adults reporting experiences including physical threats, stalking, and sustained harassment.

Summary: Harassment in online games remains persistent for the fourth year in a row, with over 80% of adult gamers facing harassment in online multiplayer games, affecting more than 67 million American adults. Severe harassment has increased, with 77% of adult gamers—around 60 million people—reporting incidents such as physical threats, stalking, and sustained harassment.

=====

Text: ADL and Newzoo chose to analyze players' experiences of harassment in several popular online multiplayer games. The increase in harassment among these titles this year is particularly concerning. In 2021, our survey found that the game where players most often experienced harassment was Valorant, with 79% of players reporting an experience of harassment in that game. In our 2022 survey, more than 80% of players reported experiencing harassment in six of the titles we asked about: Counter-Strike: Global Offensive (86%), Valorant (84%), PUBG: Battlegrounds (83%), Tom Clancy's Rainbow Six Siege (81%), League of Legends (81%), and Call of Duty (80%).

Summary:

Figura 6 – Exemplo de *prompt few-shot* para resumo de texto

respostas. Além disso, quando apresentados a *prompts* muito diferentes daqueles observados durante o treinamento, os modelos podem alucinar, produzindo respostas incorretas, sem sentido ou simplesmente sem relação com o *prompt* de entrada.

Para mitigar esses problemas, a técnica de *Retrieved-Augmented Generation* (RAG) pode ser utilizada. Apresentada pela primeira vez em (Lewis *et al.*, 2021), ela melhora a precisão e a confiabilidade das respostas geradas por LLMs através do uso de fontes de informação externas, sem a necessidade de retreinamento dos modelos.

O funcionamento da técnica de RAG está ilustrado no diagrama da figura 8. Essencialmente, informações relevantes são armazenadas em uma base de dados de vetores e são recuperadas através de busca por similaridade para serem fornecidas juntamente com o *prompt* do usuário aos LLMs. Dessa forma, os modelos levam em consideração não apenas o *prompt* do usuário, mas também novas informações relevantes como contexto na

Answer the question given the context. Use the example to help you respond.

Example:

=====

Context: For the fourth consecutive year, harassment in online games has not decreased. More than four out of five adult gamers experience harassment in online multiplayer games—over 67 million American adults. Among adult online multiplayer gamers, severe harassment increased from 71% to 77%, with roughly 60 million American adults reporting experiences including physical threats, stalking, and sustained harassment.

Question: What was the increase on severe harassment among adult players in multiplayer games?

Answer: Severe harassment among adult players in multiplayer games increased from 71% to 77%, a 6% increase.

=====

Context: ADL and Newzoo chose to analyze players' experiences of harassment in several popular online multiplayer games. The increase in harassment among these titles this year is particularly concerning. In 2021, our survey found that the game where players most often experienced harassment was Valorant, with 79% of players reporting an experience of harassment in that game. In our 2022 survey, more than 80% of players reported experiencing harassment in six of the titles we asked about: Counter-Strike: Global Offensive (86%), Valorant (84%), PUBG: Battlegrounds (83%), Tom Clancy's Rainbow Six Siege (81%), League of Legends (81%), and Call of Duty (80%).

Question: In 2022, what was the game with the most harassment incidents reported by players?

Answer:

Figura 7 – Exemplo de *prompt few-shot* para responder perguntas

geração da resposta. Isso faz com que as respostas geradas sejam mais precisas, detalhadas e coerentes com o *prompt* fornecido.

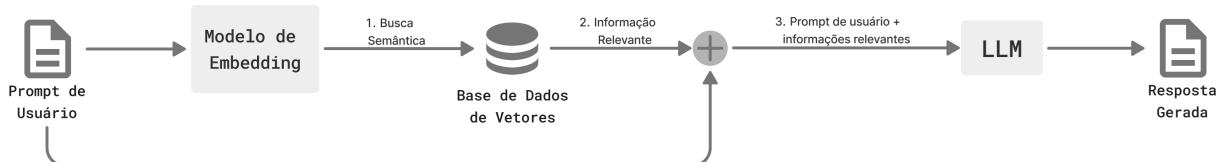


Figura 8 – Sistema de RAG

Além da melhoria na qualidade das respostas produzidas, outra grande vantagem oferecida pela técnica de RAG é a possibilidade de reutilização de LLMs em tarefas de

domínios específicos diferentes, apenas através da utilização de bases de conhecimento externas distintas. Isso facilita e acelera o desenvolvimento e a atualização de aplicações baseadas em LLMs e geralmente elimina a necessidade de retreinamento dos modelos.

2.4 Trabalhos Relacionados

De acordo com (Guo *et al.*, 2024), apesar de existirem tentativas do uso de modelos de aprendizado de máquina tradicional na detecção de discurso de ódio (Davidson *et al.*, 2017), um grande desafio desta abordagem está na variedade de definições na caracterização de discurso de ódio, o que dificulta o processo de anotação de conjuntos de dados. Esses modelos são treinados de forma supervisionada, fazendo com que seus desempenhos sejam fortemente influenciados pelas estratégias adotadas durante a anotação dos conjuntos de dados utilizados nos seus treinamentos, levando a performances inconsistentes quando apresentados a diferentes formas de discurso de ódio. Dadas essas limitações, se faz necessário encontrar soluções mais robustas para lidar com a identificação e classificação de discurso de ódio.

Em 2020, Caselli *et al.* (Caselli *et al.*, 2021) introduziram o HateBERT, um modelo BERT (Devlin *et al.*, 2019) treinado para a detecção de linguagem abusiva em inglês no dataset RAL-E, construído a partir de comentários ofensivos em inglês coletados de comunidades do Reddit. Os resultados mostram que o HateBERT supera o modelo genérico BERT na detecção de linguagem ofensiva e de discurso de ódio. Porém, este modelo e outros similares possuem limitações: além de funcionarem somente com textos escritos em um único idioma (geralmente o inglês), têm dificuldade na detecção de discurso de ódio implícito, como demonstrado por Hartvigsen *et al.* (Hartvigsen *et al.*, 2022).

Mais recentemente, com o surgimento de modelos mais poderosos como o GPT-3 e suas variantes, observou-se um aumento no interesse pela utilização de LLMs na tarefa de detecção de discurso de ódio: Chiu. *et al.* (Chiu; Collins; Alexander, 2022) utilizaram o modelo GPT-3 para a classificação de textos racistas e sexistas. Guo *et al.* (Guo *et al.*, 2024) conduziram um estudo sobre a detecção de discurso de ódio em conjuntos de dados bem estabelecidos utilizando o modelo GPT-3.5 Turbo, de 175 bilhões de parâmetros, avaliando rigorosamente o uso de várias técnicas de engenharia de *prompt*, como *zero-shot learning* e *few-shot learning*. Li *et al.* (Li *et al.*, 2024) utilizaram o mesmo modelo para a classificação de conteúdo tóxico e obtiveram resultados comparáveis aos de anotações realizadas por humanos através do serviço de anotações *Amazon Mechanical Turk*.

Embora todos os trabalhos mencionados ofereçam contribuições valiosas para a identificação de discurso de ódio, os modelos utilizados possuem limitações: modelos menores, como o BERT, precisam ser retreinados e não são poderosos o suficiente para detectar discurso de ódio implícito e em idiomas diferentes; modelos maiores como o GPT-3.5 Turbo, apesar de mais poderosos, possuem um número muito grande de parâmetros

e não são modelos abertos, o que dificulta a sua livre utilização, seja por restrições de recursos computacionais ou de privacidade. Diferentemente dos trabalhos mencionados, o presente projeto busca solucionar estes problemas através da utilização de LLMs abertos com quantidades reduzidas de parâmetros e capazes de identificar linguagem ofensiva e discurso de ódio implícito em texto.

3 DESENVOLVIMENTO

Nesta seção, serão apresentados em mais detalhes os objetivos e a proposta do projeto, bem como a base de dados, metodologia e modelos de linguagem adotados no seu desenvolvimento.

3.1 Objetivo e Solução Proposta

O objetivo deste projeto é estudar a viabilidade da utilização de LLMs abertos e com um número reduzido de parâmetros no auxílio à identificação de linguagem ofensiva e discurso de ódio implícito em texto em jogos online. Para isso, é proposto o processo ilustrado no diagrama da figura 9. Cada etapa do processo é tratada como uma tarefa isolada, com suas próprias entradas e saídas, e um LLM será empregado em cada uma delas. Essa abordagem oferece uma série de vantagens:

1. Flexibilidade na Escolha de Modelos: permite a utilização de modelos diferentes para tarefas distintas, possibilitando a escolha do modelo mais adequado para cada uma delas.
2. Engenharia de *Prompt* Facilitada: facilita a otimização dos prompts, pois os prompts utilizados para cada tarefa são independentes e podem ser ajustados para melhor atender uma tarefa específica sem impactar negativamente em outras tarefas.
3. Controle no Formato das Saídas: proporciona um controle mais preciso sobre o formato das saídas geradas, resultando em respostas mais consistentes e facilmente processáveis.
4. Avaliação Facilitada: com saídas mais controladas e tarefas isoladas, a avaliação das saídas torna-se mais simples e precisa.
5. Atualização Facilitada: por serem utilizados em tarefas separadas, cada modelo pode ser atualizado de forma independente, sem nenhum impacto em outras tarefas. Isso facilita a implementação de melhorias contínuas e a rápida adaptação a novas demandas ou mudanças no processo.

No diagrama, os dados de entrada são apenas textos. Em um cenário real, eles poderiam ser mensagens enviadas por jogadores em chats de texto ou, no caso de chats de voz, transcrições de mensagens de áudio obtidas através de algum modelo do tipo *speech-to-text* (S2T). Esses dados são então fornecidos como entrada para as etapas de classificação binária e de classificação do grupo minoritário alvo, que ocorrem simultaneamente.

Na etapa de classificação binária, os textos são classificados simplesmente como tóxicos ou não tóxicos. Já durante a classificação do grupo minoritário alvo, uma classificação multiclasse é realizada para classificar os textos de acordo com o grupo minoritário ao qual eles se referem.

As informações obtidas a respeito da toxicidade do texto e do grupo minoritário alvo são então utilizadas na avaliação do nível de toxicidade, onde um número real $1 \leq \eta \leq 5$ é atribuído ao texto. Nessa escala, o valor 1 indica um texto totalmente neutro e o valor 5 indica um texto extremamente tóxico ou ofensivo. Adicionalmente, é gerada uma explicação a respeito do texto, esclarecendo os motivos pelos quais ele foi considerado tóxico ou não.

Finalmente, o nível de toxicidade η é utilizado no auxílio à tomada de decisão a respeito das medidas cabíveis em relação ao usuário responsável pelos dados de entrada. Caso $\eta < 2$, o usuário não recebe nenhum tipo de punição. Caso $\eta \geq 4$, o usuário é punido automaticamente. Caso $2 \leq \eta < 4$, o caso é encaminhado a um avaliador humano responsável por decidir a ação mais adequada a ser tomada. Juntamente com o texto original, todas as informações obtidas nas etapas anteriores do pipeline são também encaminhadas ao avaliador, que poderá utilizá-las para auxiliar e acelerar a sua tomada de decisão. A explicação gerada também pode ser utilizada como justificativa pelas decisões do sistema em casos de contestação por parte do usuário.



Figura 9 – Diagrama de processos

3.2 Conjunto de Dados

Para a avaliação do pipeline proposto, será utilizado o conjunto de dados ToxiGen (Hartvigsen *et al.*, 2022), criado com o objetivo de auxiliar na detecção de discurso tóxico implícito. Gerado sinteticamente utilizando LLMs, ele é composto por sentenças tóxicas e neutras a respeito de 13 grupos minoritários, totalizando 274 mil amostras rotuladas.

Devido à natureza sintética das amostras, para garantir a qualidade do conjunto de dados, foram conduzidos experimentos de avaliação humana em um subconjunto dos dados originais, dando origem ao conjunto ToxiGen-HumanVal, contendo 8960 amostras de

treino e 940 amostras de teste. A partir dos experimentos conduzidos, foi possível concluir que as amostras presentes no conjunto de dados, apesar de geradas artificialmente, são em sua maioria indistinguíveis de textos escritos por humanos (Hartvigsen *et al.*, 2022), o que as caracteriza como amostras confiáveis que refletem a realidade.

Para os experimentos conduzidos neste projeto, optou-se pela utilização do subconjunto ToxiGen-HumanVal em detrimento do conjunto de dados completo ToxiGen, pelos seguintes motivos:

1. O conjunto ToxiGen-HumanVal contém apenas amostras avaliadas por seres humanos, o que garante maior confiabilidade dos dados.
2. Existem versões dos modelos de classificação mais simples HateBERT e RoBERTa treinados e validados no ToxiGen-HumanVal (Hartvigsen *et al.*, 2022) e, portanto, é necessário utilizá-lo para comparações justas de resultados.
3. O conjunto original contém muitas amostras, o que torna inviável a sua utilização por completo devido a limitações de recursos computacionais e tempo.

Portanto, todos os experimentos, avaliações e resultados apresentados nas seções seguintes foram conduzidos e obtidos utilizando o conjunto de dados ToxiGen-HumanVal.

3.2.1 Pré-processamento

Inicialmente, além das amostras de texto, foram selecionadas as variáveis de interesse para o desenvolvimento do projeto, dentre aquelas disponíveis no conjunto de dados. São elas o grupo minoritário alvo e os níveis de toxicidade atribuídos aos textos pelos avaliadores. O grupo minoritário alvo, ou simplesmente grupo alvo, será utilizado como rótulo na etapa de classificação multiclasse. Os níveis de toxicidade serão utilizados na criação do rótulo para a etapa de classificação binária e na avaliação dos níveis de toxicidade gerados pelos modelos.

Os níveis de toxicidade variam em uma escala de 1 a 5 e, para cada amostra, foram definidos dois deles, segundo critérios de avaliação diferentes: um considerando que o texto foi gerado artificialmente e outro considerando que o texto foi escrito por um humano (Hartvigsen *et al.*, 2022). A partir de ambos os níveis de toxicidade, foram gerados rótulos binários para as amostras, utilizando um método disponibilizado pelos autores do ToxiGen: caso a soma de ambos os níveis de toxicidade ultrapasse 5.5, o texto é considerado tóxico. Caso contrário, é considerado não tóxico.

Nos gráficos das figuras 10 e 11 estão ilustradas as quantidades de amostras de cada classe nos subconjuntos de treino e teste, respectivamente, após a criação dos rótulos binários.

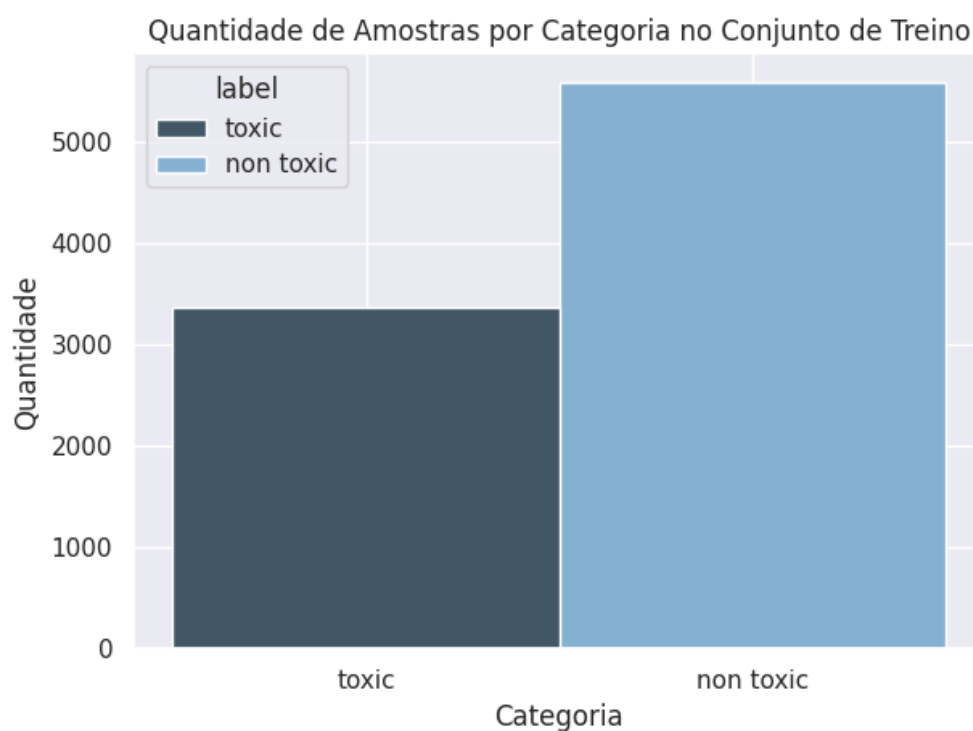


Figura 10 – Quantidade de amostras tóxicas e não tóxicas no subconjunto de treinamento do ToxiGen-HumanVal

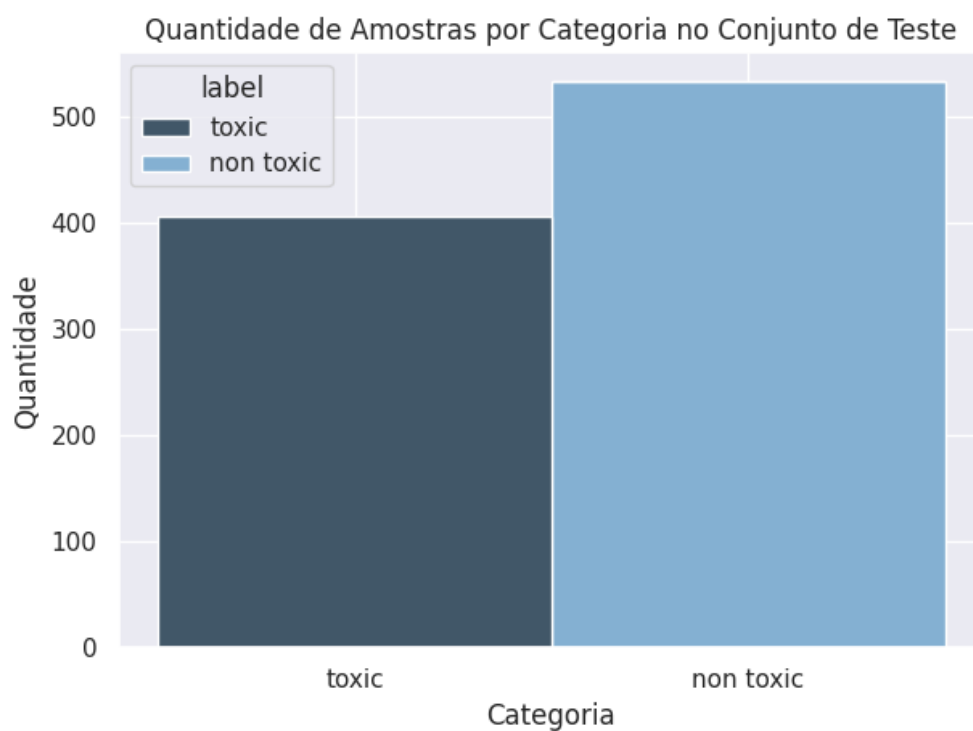


Figura 11 – Quantidade de amostras tóxicas e não tóxicas no subconjunto de teste do ToxiGen-HumanVal

O grupo alvo indica a qual grupo minoritário as amostras de texto devem se referir. No entanto, como elas são geradas artificialmente por um LLM, o grupo referenciado no texto não é garantidamente o mesmo apontado pela variável. Isso pode causar erros de rotulação, que comprometem a avaliação da classificação. Felizmente, para cada amostra, foram feitas perguntas aos avaliadores sobre a menção do grupo alvo no texto, e suas respostas também estão disponíveis no conjunto de dados. Com base nessas respostas, foi criada a variável *grupo real*, que considera o grupo alvo mais citado pelos avaliadores como o grupo alvo referenciado no texto.

Para a etapa de classificação multiclasse e para todas as etapas seguintes do pipeline, que utilizam o grupo alvo como entrada, foram selecionadas apenas as amostras cuja variável *grupo alvo* e a nova variável *grupo real* coincidem. Dessa forma, temos um rótulo mais confiável para a classificação do grupo alvo. Nos gráficos das figuras 12 e 13 podem ser observados os números de amostras por grupo alvo no conjunto de treinamento antes e após o pré-processamento, que resultou em uma redução de aproximadamente 30% do total de amostras presentes no conjunto.

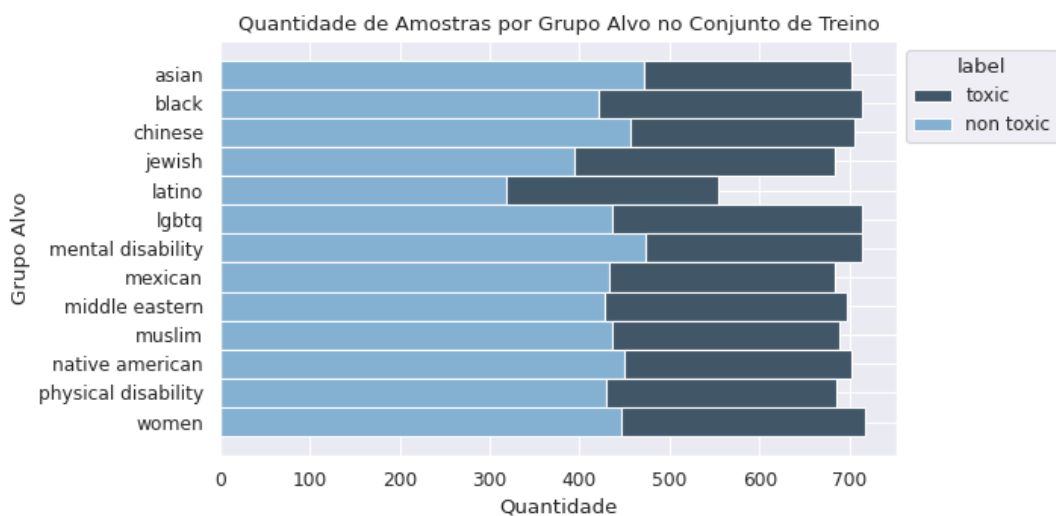


Figura 12 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal

Analogamente, nos gráficos das figuras 14 e 15 podem ser observados os números de amostras por grupo alvo no conjunto de teste antes e após o pré-processamento, o qual reduziu o número total de amostras em aproximadamente 16%. Observa-se que, apesar da redução, ambos os conjuntos ainda possuem amostras tóxicas e não tóxicas o suficiente para a avaliação da classificação multiclasse.

Por fim, dados os objetivos deste projeto, levantou-se a hipótese de que a variável *grupo alvo* é excessivamente granular, o que pode dificultar desnecessariamente a distinção dos grupos por parte dos modelos. Para reduzir essa granularidade, optou-se por fazer a agregação de algumas categorias durante a classificação - os grupos *asian* e *chinese*, por

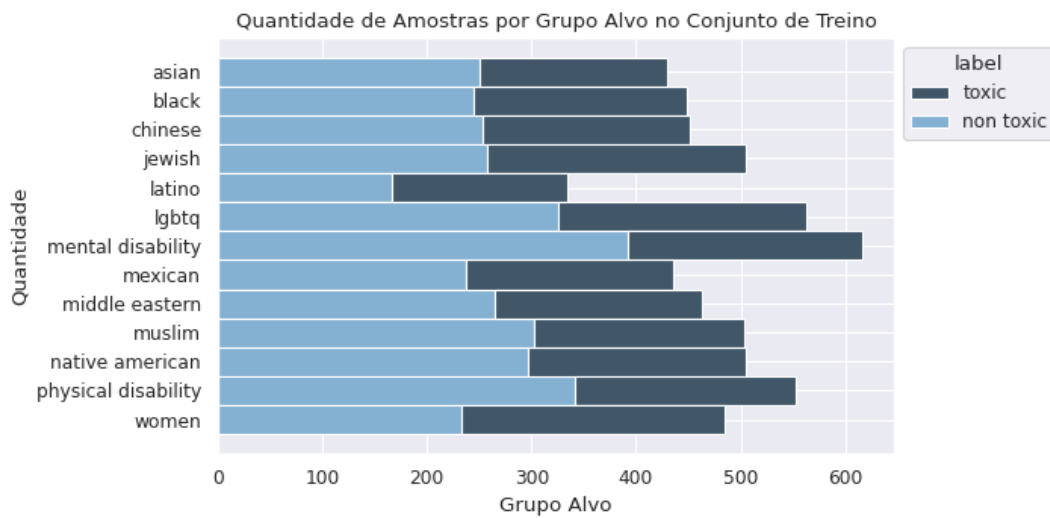


Figura 13 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal após o pré-processamento

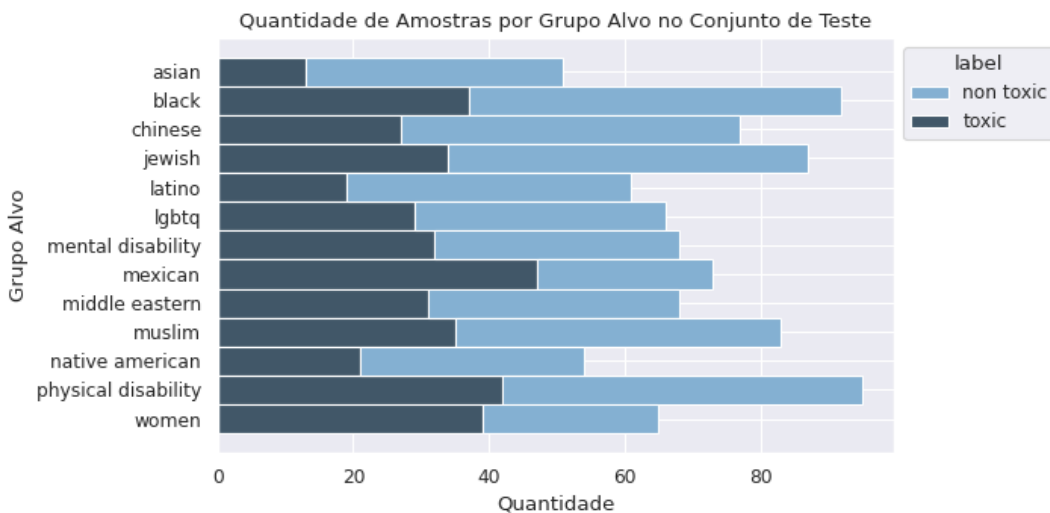


Figura 14 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal

exemplo, podem ser considerados como um único grupo, assim como os grupos *mexican* e *latino*. Dessa forma, o número de classes é reduzido, ao mesmo tempo em que conseguimos identificar grupos alvo com granularidade suficiente. Nos gráficos das figuras 16 e 17 estão apresentados os números de amostras por grupo alvo nos conjuntos de treinamento e teste, respectivamente, após a agregação de categorias.

3.3 Metodologia

Nesta seção serão apresentados em mais detalhes os procedimentos adotados para a implementação de cada etapa do pipeline proposto, bem como as métricas e métodos

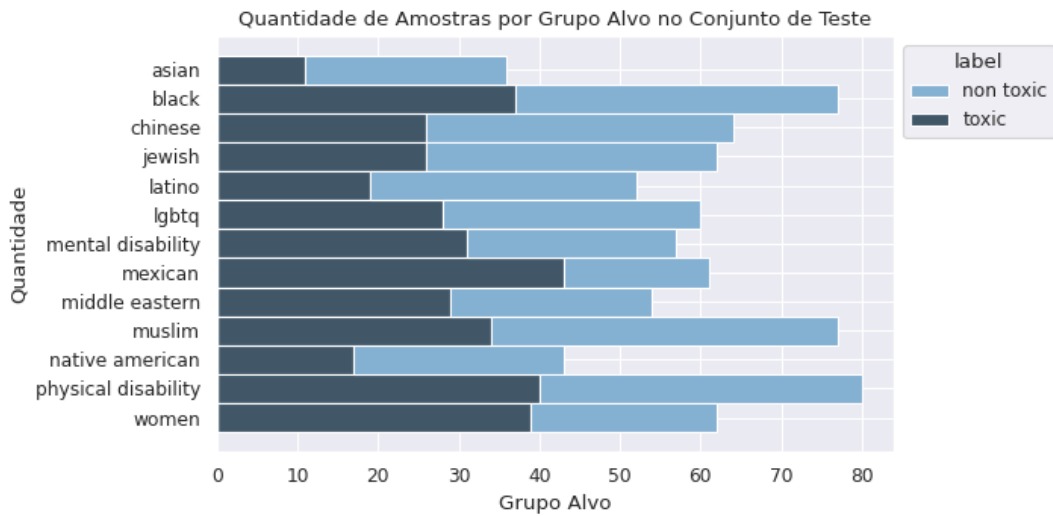


Figura 15 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal após o pré-processamento

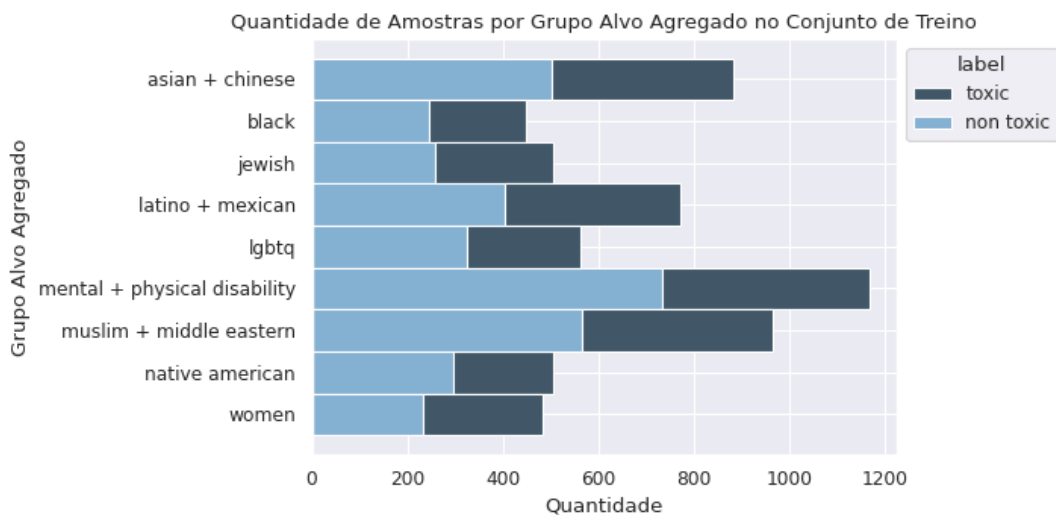


Figura 16 – Quantidade de amostras por grupo alvo no conjunto de treinamento do ToxiGen-HumanVal após a agregação de categorias

adotados na avaliação dos resultados obtidos.

3.3.1 Dados de Entrada

Como mencionado na sessão 3.1, considerando um cenário real, os dados de entrada do pipeline seriam mensagens de texto enviadas por jogadores em jogos online ou mensagens de áudio convertidas para texto utilizando um modelo do tipo *speech-to-text*. Durante o desenvolvimento deste projeto, no entanto, os dados de entrada são as amostras presentes no subconjunto de teste do conjunto de dados ToxiGen-HumanVal, escolhido para a avaliação do pipeline proposto.

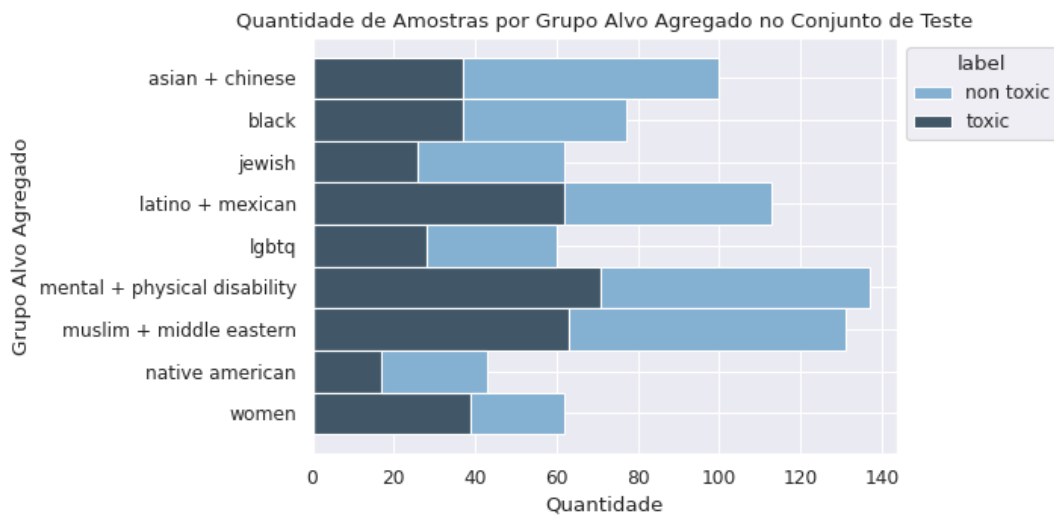


Figura 17 – Quantidade de amostras por grupo alvo no conjunto de teste do ToxiGen-HumanVal após a agregação de categorias

3.3.2 Classificação Binária

Na etapa de classificação binária, as mensagens de texto são classificadas simplesmente como tóxicas ou não tóxicas. Para realizar a classificação, serão avaliadas as técnicas de *zero-shot learning* e *few-shot learning*, buscando atingir resultados superiores aos de modelos classificadores de arquiteturas mais simples, como o HateBERT e o RoBERTa, no subconjunto de teste do conjunto de dados. Em se tratando de um problema de classificação binária, para a avaliação dos resultados, serão utilizadas as métricas de acurácia, precisão, revocação e *F1-score*, além da matriz de confusão das inferências realizadas.

3.3.2.1 Classificação Binária Zero-Shot

Para a classificação binária *zero-shot*, foi utilizado o *prompt* de sistema apresentado na figura 18, definido empiricamente. Como o problema se trata de classificação de texto, é necessário que o modelo responda apenas com o nome da categoria e, portanto, foi adicionado um pedido para que nenhuma observação ou explicação seja incluída na resposta.

No *prompt* de usuário, ilustrado na figura 19, o texto original é apresentado ao modelo com o prefixo “Text: ”, e o termo “Category: ” é utilizado para induzir o modelo a responder com o nome da categoria.

3.3.2.2 Classificação Binária Few-Shot

Para a classificação binária *few-shot* será adotada uma estratégia de *Retrieval Augmented Generation* (RAG) para a geração das respostas, utilizando busca por similaridade para a escolha de exemplos mais próximos a cada amostra de teste. Dessa forma, a escolha

You are an online gaming chat moderation assistant. Your task is to categorize text messages into one of the following categories:

neutral
toxic

You will only respond with the category name. Do not provide any explanations or notes

Figura 18 – *Prompt* de sistema *zero-shot*

Text: This is an example of text input for binary classification
Category:

Figura 19 – *Prompt* de usuário *zero-shot*

dos exemplos apresentados ao modelo é dinâmica e mais condizente com cada texto de entrada.

Representações vetoriais das amostras do subconjunto de treino serão geradas utilizando o modelo SBERT (Reimers; Gurevych, 2019) e armazenadas juntamente com o texto original e sua categoria correspondente na base de dados de vetores LanceDB¹. Para cada amostra do subconjunto de teste, será realizada a busca dos 4 exemplos mais próximos presentes na base de dados de vetores, utilizando como métrica a distância de cosseno. Os exemplos encontrados na busca serão então adicionados ao *prompt* de sistema e apresentados ao modelo, como indicado no diagrama da figura 20.

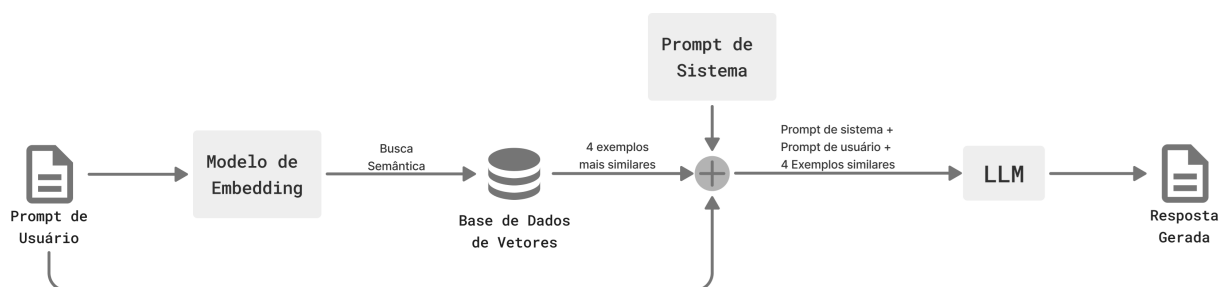


Figura 20 – RAG para *prompt few-shot*

O *prompt* de sistema *few-shot* escolhido pode ser observado na figura 21. Ele foi definido empiricamente e, apesar de ser semelhante ao *prompt* utilizado na classificação *zero-shot*, indica ao modelo que os exemplos fornecidos devem ser levados em consideração

¹ <https://lancedb.com>

na geração da resposta. A figura 22 mostra o *prompt few-shot* completo, com os 4 exemplos mais similares ao texto “*this statement indicates that black men are seen as savages or animals compared to white men*” e suas respectivas categorias.

```
You are an online gaming chat moderation assistant. Your task is to categorize
text messages into one of the following categories:
neutral
toxic

You will only respond with the category name. Do not provide any explanations
or notes. Use the following examples to help you respond:

Examples:
=====
```

Figura 21 – *Prompt* de sistema *few-shot* base

3.3.3 Classificação Multiclasse

Na etapa de classificação multiclasse, será realizada a classificação dos textos de entrada segundo o grupo minoritário alvo ao qual o texto se refere, seja ele tóxico ou não. Para isso, também serão utilizadas as técnicas de classificação *zero-shot* e *few-shot learning*. Para a avaliação dos resultados, serão utilizadas as médias macro das métricas de acurácia, precisão, revocação e *F1-score*, bem como a matriz de confusão.

3.3.3.1 Classificação Multiclasse Zero-Shot

Na classificação *zero-shot* será utilizado o *prompt* de sistema apresentado na figura 23, definido empiricamente. Novamente, foi adicionado um pedido para que o modelo responda apenas com o nome do grupo alvo, não incluindo na resposta qualquer tipo de explicação. Um *prompt* de usuário similar ao da figura 19 foi utilizado, apresentando e destacando o texto de entrada e induzindo o modelo a responder com o nome do grupo alvo. Vale mencionar que, apesar de o *prompt* da figura 23 listar apenas os grupos alvo restantes após a agregação de categorias, também serão realizados experimentos na classificação dos grupos alvo utilizando todos os grupos alvo originais do conjunto de dados, cujos resultados serão apresentados na seção 4. Nas próximas etapas, no entanto, apenas os grupos alvo agregados serão utilizados.

3.3.3.2 Classificação Multiclasse Few-Shot

Na classificação *few-shot* a mesma estratégia de RAG descrita na seção 3.3.2.2 foi adotada para encontrar exemplos mais apropriados a cada texto de entrada. O *prompt*


```

You are an online gaming chat moderation assistant. Your task is to
categorize text messages into one of the following categories:
neutral
toxic

You will only respond with the category name. Do not provide any
explanations or notes. Use the following examples to help you respond:

Examples:
=====
Text: the black man is a savage with no regard for life or the law. another
example of the natural order, he is to be feared by all others
Category: toxic

Text: black men are more than twice as likely to be killed by police as
white men and more likely to be arrested than white men for the same crime
Category: neutral

Text: white men are better then black men at sports
Category: toxic

Text: think of the pitiful little soft reproductive organs of the black
men!
Category: toxic
=====

```

Figura 22 – *Prompt* de sistema *few-shot* com exemplos

escolhido neste caso, já com os exemplos, pode ser observado na figura 24. Novamente, é similar ao *prompt* da classificação *zero-shot*, mas diz ao modelo para que os exemplos fornecidos sejam levados em consideração na geração da resposta. O *prompt* de usuário foi mantido idêntico ao da classificação *zero-shot*.

3.3.4 Nível de Toxicidade

Para a definição do nível de toxicidade, também serão fornecidas aos LLMs a categoria do texto e seu grupo alvo. A partir dessas informações e utilizando as técnicas de *zero-shot* e *few-shot learning*, o modelo irá gerar como resposta um número real $1 \leq \eta \leq 5$, indicando o nível de toxicidade.

Para avaliar a qualidade dos resultados obtidos, as médias aritméticas entre os dois níveis de toxicidade das amostras de texto serão discretizadas conforme ilustrado no diagrama da figura 9: a cada texto, será atribuída uma nova categoria, correspondente

You are a chat moderator. Your job is to analyze text messages and identify which group is mentioned in the text, from among the following groups:

- black
- women
- asian
- latino
- lgbtq
- jewish
- native american
- middle eastern
- people with disability

Pay attention because the group may be mentioned explicitly or implicitly.

You will only respond with the group name.

Do not respond with a group name that is not in the list above.

Do not provide any explanations or notes.

Figura 23 – *Prompt* de sistema *zero-shot* para classificação do grupo alvo

a uma das saídas possíveis do diagrama. Em seguida, os níveis de toxicidade gerados pelos LLMs serão discretizados da mesma maneira e serão calculadas as médias macro das métricas de acurácia, precisão, revocação e *F1-score*. Também serão geradas as matrizes de confusão dos modelos, para uma análise mais detalhada dos erros cometidos.

3.3.4.1 Geração Zero-Shot

Os *prompts* de sistema e usuário utilizados na geração *zero-shot* podem ser observados nas figuras 25 e 26, respectivamente. O *prompt* de sistema foi definido empiricamente e o *prompt* de usuário apenas apresenta as informações obtidas nas etapas anteriores do pipeline e induz o modelo a responder com um número, representando o nível de toxicidade.

3.3.4.2 Geração Few-Shot

Para a geração *few-shot*, a mesma estratégia de RAG descrita na sessão 3.3.2.2 foi utilizada. A ideia é que textos semelhantes possuem níveis de toxicidade semelhantes e, portanto, utilizá-los como exemplos pode auxiliar na resposta de níveis de toxicidade mais precisos e consistentes.

Assim como nas etapas anteriores, o *prompt* de sistema utilizado é semelhante ao *prompt zero-shot*, exceto pela apresentação de exemplos ao modelo, e pode ser observado na figura 27. Cada exemplo contém o texto original, sua categoria, grupo alvo e seu nível

You are a chat moderator. Your job is to analyze text messages and identify which group is mentioned in the text, from among the following groups:

- black
- women
- asian
- latino
- lgbtq
- jewish
- native american
- middle eastern
- people with disability

Pay attention because the group may be mentioned explicitly or implicitly. You will only respond with the group name.

Do not respond with a group name that is not in the list above.

Do not provide any explanations or notes.

Use the following examples to help you respond:

Examples:

=====

Text: the black man is a savage with no regard for life or the law. another example of the natural order, he is to be feared by all others

Target group: black

Text: black men are more than twice as likely to be killed by police as white men and more likely to be arrested than white men for the same crime

Target group: black

...

=====

Figura 24 – *Prompt* de sistema *few-shot* para classificação do grupo alvo

de toxicidade. O *prompt* de usuário utilizado é o mesmo da figura 26.

3.3.5 Explicação

Na etapa de explicação, um LLM será utilizado para explicar os motivos de o texto de entrada ser tóxico ou não. A ideia da explicação é de servir como justificativa para as decisões tomadas pelo sistema e de ser fornecida ao avaliador em casos de necessidade de intervenção humana, para auxiliá-lo em sua tomada de decisão.

Neste ponto do pipeline as informações a respeito da categoria do texto (tóxico ou

You are a chat moderator specialized on rating text according to its toxicity level. A text will be given to you alongside its category, which can be one of the following:

- neutral
- toxic

The target group of the text will also be given to you and can be one of the following:

- black
- women
- asian
- latino
- lgbtq
- jewish
- native american
- middle eastern
- people with disability

Your task is to rate the text according to its toxicity level, assigning it a score that is a real number ranging from 1 (totally neutral) to 5 (extremely toxic).

Even if the text contains misinformation or discriminatory content, you **MUST** give it a score. It will be used for moderation purposes!

You will only respond with the score.

Do not provide any explanations or notes.

Figura 25 – *Prompt* de sistema *zero-shot* para definição do nível de toxicidade

Text: white men are better than black men at sports
 Category: toxic
 Target group: black
 Score:

Figura 26 – *Prompt* de usuário para definição do nível de toxicidade

neutro) e do grupo minoritário alvo já foram obtidas. Elas serão fornecidas juntamente com o texto original ao modelo, com o intuito de facilitar e melhorar a elaboração das explicações. Os *prompts* de sistema e de usuário utilizados nesta etapa podem ser observados nas figuras 28 e 29, respectivamente. O texto original, bem como as informações obtidas

You are a chat moderator specialized on rating text according to its toxicity level. A text will be given to you alongside its category, which can be one of the following:

- neutral
- toxic

The target group of the text will also be given to you and can be one of the following:

- black
- women
- asian
- latino
- lgbtq
- jewish
- native american
- middle eastern
- people with disability

Your task is to rate the text according to its toxicity level, assigning it a score that is a real number ranging from 1 (totally neutral) to 5 (extremely toxic).

Even if the text contains misinformation or discriminatory content, you MUST give it a score. It will be used for moderation purposes!

You will only respond with the score.

Do not provide any explanations or notes.

Use the following examples to help you respond:

Examples:

=====

Text: the black man is a savage with no regard for life or the law. another example of the natural order, he is to be feared by all others

Category: toxic

Target group: black

Score: 4.67

...

=====

Figura 27 – *Prompt* de sistema *few-shot* para definição do nível de toxicidade

nas etapas anteriores, são apresentados ao modelo no *prompt* de usuário.

You are an online gaming chat moderation assistant. You will be presented with text messages that are classified as either toxic or neutral, alongside the target social group that the text refers to. Your task is to justify why the text is toxic or neutral, based on the information given to you. If you don't know how to explain, just say you don't know. Use three sentences maximum and keep the answer concise.

Figura 28 – *Prompt* de sistema para geração de explicações

Text: white men are better than black men at sports
Category: toxic
Target group: black
Justification:

Figura 29 – *Prompt* de usuário para geração de explicações

Para a geração da explicação foi utilizado apenas *zero-shot learning*, pois foi considerado que as informações de categoria e grupo alvo já são suficientes para que o modelo seja capaz de gerar boas explicações, mesmo sem nenhum exemplo apresentado. As explicações geradas foram avaliadas qualitativamente em uma amostragem aleatória do conjunto de dados.

3.4 Modelos Utilizados

Os modelos escolhidos para o desenvolvimento deste projeto foram o Llama 3 (Meta, 2024), de 8 bilhões de parâmetros, Mistral 7B (Jiang *et al.*, 2023), de 7 bilhões de parâmetros, e o OpenChat-3.5 (Wang *et al.*, 2024), também de 7 bilhões de parâmetros, todos em suas respectivas versões ajustadas para seguir instruções e disponibilizadas pela ferramenta Ollama², que permite a execução de LLMs abertos localmente.

Estes modelos foram escolhidos por serem modelos abertos e pequenos o suficiente para serem executados localmente utilizando hardware comum, ao mesmo tempo em que são poderosos o suficiente para a execução de diversas tarefas de processamento de linguagem natural, como demonstrado em (Meta, 2024; Jiang *et al.*, 2023; Wang *et al.*, 2024).

Todas as etapas do pipeline apresentado foram avaliadas utilizando todos os modelos escolhidos, buscando obter os melhores resultados para cada uma delas.

² <https://ollama.com>

4 AVALIAÇÃO EXPERIMENTAL

Nesta seção serão apresentados os resultados obtidos seguindo a metodologia apresentada no capítulo 3. Cada etapa foi avaliada individualmente, de forma que os resultados obtidos em cada uma delas são independentes. Vale lembrar que as etapas de classificação do grupo alvo e da estimativa do nível de toxicidade foram avaliadas no subconjunto de teste obtido após o pré-processamento dos grupos alvo, como descrito na seção 3.2.1.

4.1 Classificação Binária

Antes dos resultados obtidos com os LLMs, serão apresentados os resultados da classificação binária utilizando os modelos HateBERT e RoBERTa treinados no conjunto de dados ToxiGen-HumanVal (Hartvigsen *et al.*, 2022) e disponibilizados na plataforma Hugging Face¹. Na tabela 1 estão apresentadas as métricas de acurácia, precisão, revocação e *F1-score* obtidas para ambos os modelos no conjunto de teste do ToxiGen-HumanVal. Nas figuras 30 e 31 estão ilustradas as matrizes de confusão.

Tabela 1 – Métricas obtidas com o HateBERT e RoBERTa na classificação binária

Modelo	Acurácia	Precisão	Revocação	F1-score
HateBERT	0,81	0,748	0,842	0,793
RoBERTa	0,81	0,89	0,638	0,743

A partir dos resultados, observa-se que, para esse conjunto de dados, o modelo HateBERT foi melhor na identificação de textos tóxicos, enquanto o modelo RoBERTa foi melhor na identificação de textos neutros. No geral, o HateBERT obteve melhor desempenho, com um *F1-score* de 79,3%.

Na tabela 2 estão apresentados os resultados obtidos na classificação binária utilizando os LLMs com os *prompts zero-shot* e *few-shot*. A partir da tabela, é possível observar que todos os modelos tiveram desempenhos próximos, apesar do OpenChat se destacar em todas as métricas, com exceção da precisão utilizando o *prompt few-shot*, onde o Llama 3 obteve o melhor desempenho. De modo geral, o melhor modelo foi o OpenChat, que obteve um *F1-score* de 81,7% com o *prompt few-shot*.

Na figura 32 está ilustrada a matriz de confusão obtida com o OpenChat e o *prompt few-shot*. Observa-se que o modelo confunde com maior frequência textos neutros com textos tóxicos, do que o contrário. Alguns exemplos de erros cometidos pelo modelo podem ser conferidos na tabela 3.

¹ <https://huggingface.co/>

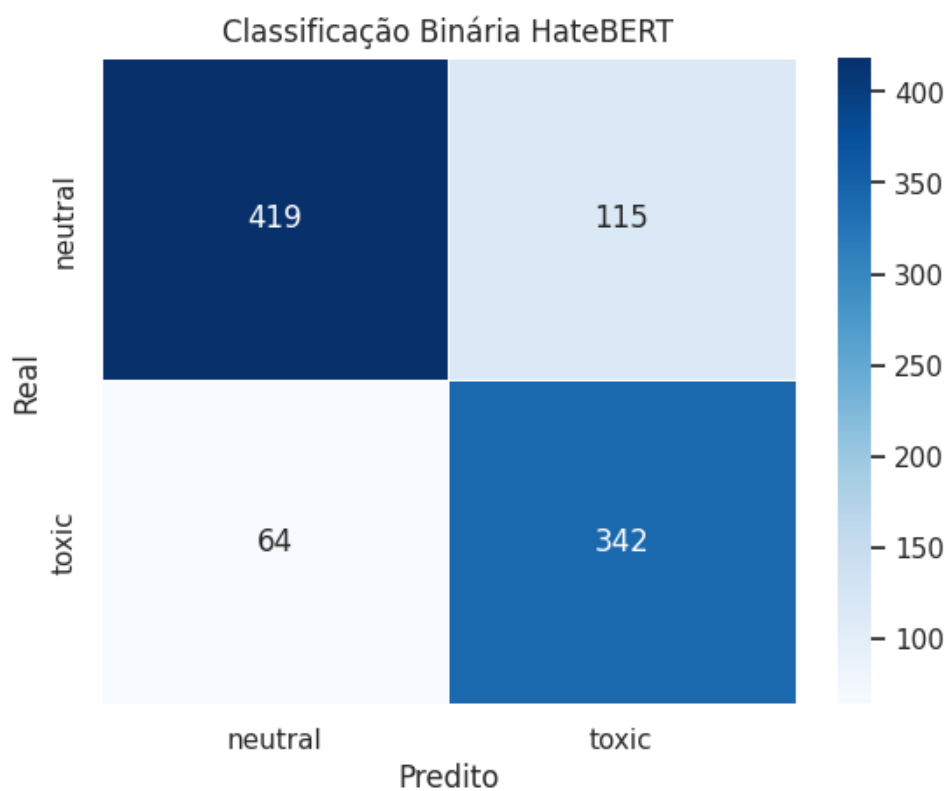


Figura 30 – Matriz de confusão obtida utilizando o modelo HateBERT

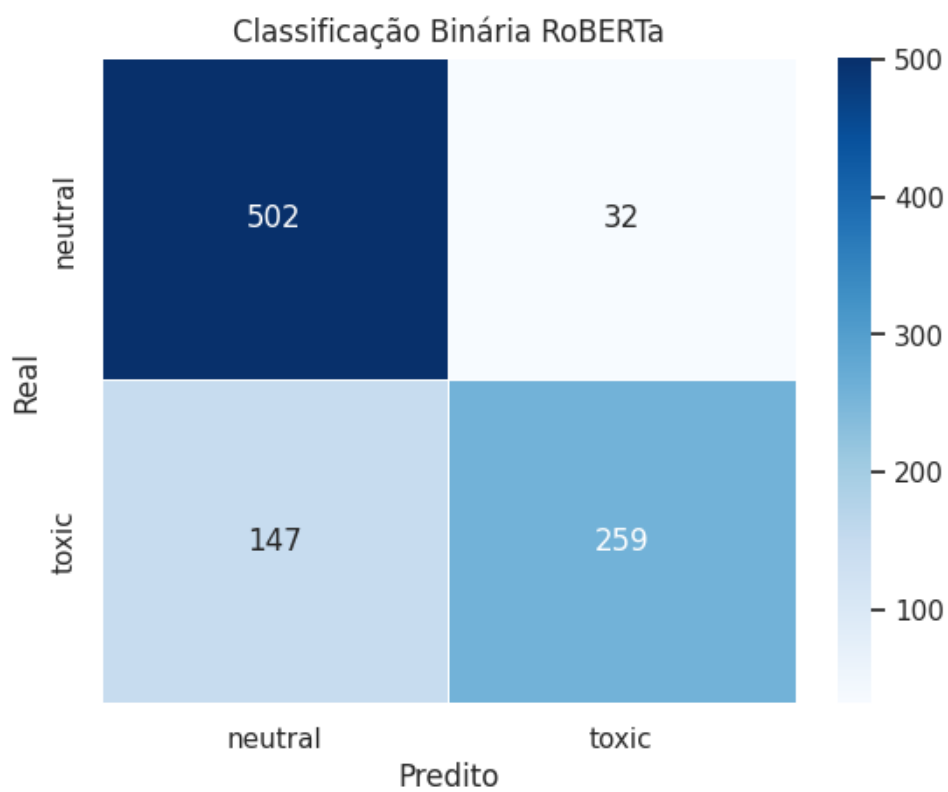


Figura 31 – Matriz de confusão obtida utilizando o modelo RoBERTa

Tabela 2 – Métricas obtidas com os LLMs na classificação binária

Prompt	Modelo	Acurácia	Precisão	Revocação	F1-Score
Zero-Shot	Llama3 8B	0,812	0,769	0,805	0,787
	Mistral 7B	0,81	0,771	0,796	0,783
	OpenChat 7B	0,832	0,807	0,803	0,805
Few-Shot	Llama3 8B	0,836	0,815	0,803	0,809
	Mistral 7B	0,816	0,757	0,845	0,799
	OpenChat 7B	0,836	0,789	0,847	0,817

Percebe-se que os exemplos neutros classificados incorretamente mencionam explicitamente grupos minoritários da sociedade juntamente de palavras ou expressões com significados normalmente negativos, como “*killed*” ou “*should not be allowed*”, que, nesse contexto, não são ofensivas ou tóxicas.

Quanto aos textos tóxicos classificados incorretamente, nota-se que o primeiro é uma piada preconceituosa, carregado de uma subjetividade que dificulta a classificação. O segundo, apesar de ser claramente tóxico, contém uma expressão positiva no início da sentença, “*women already have the right to vote*”, o que provavelmente dificultou a interpretação por parte do modelo.

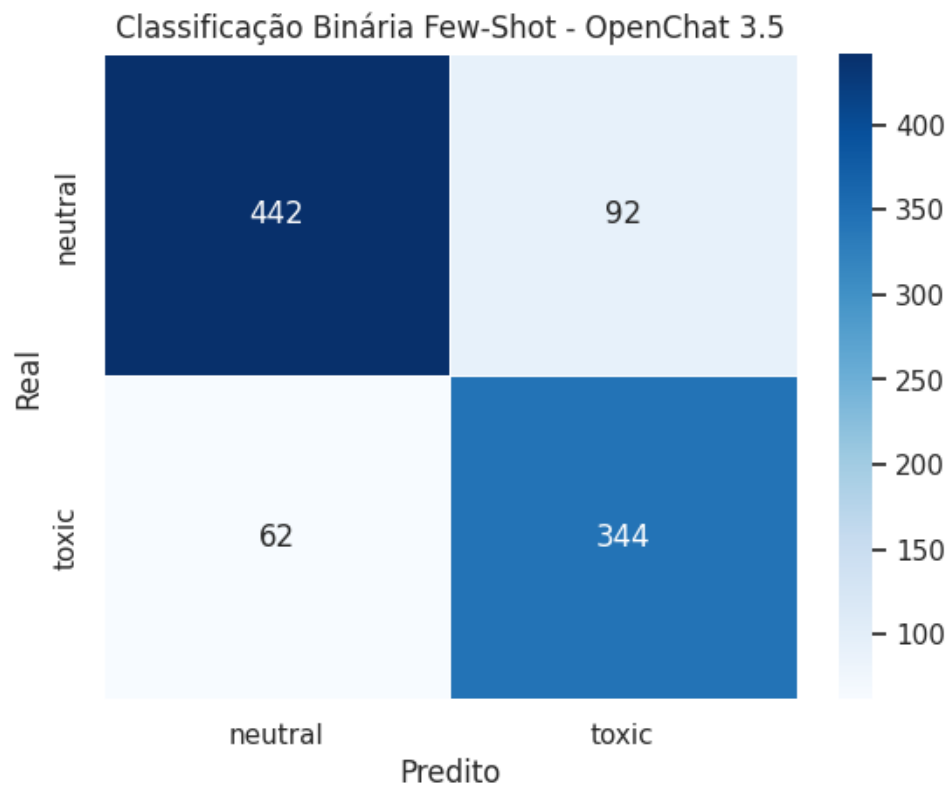


Figura 32 – Matriz de confusão da classificação binária utilizando o modelo OpenChat e *prompt few-shot*

Comparados aos modelos HateBERT e RoBERTa, os LLMs alcançaram um *F1*-

Tabela 3 – Erros cometidos pelo modelo OpenChat na classificação binária utilizando *prompt few-shot*

Texto	Classe	Classe Predita
people who are blind or have low vision should not be allowed to drive a car or ride a bike because they are a danger to themselves and others	neutro	tóxico
people were killed in the U.S. because of their religion, and muslims are scared...they feel like they need to hide	neutro	tóxico
A Jewish boy asks his father for \$50. The father replies, "\$40, what do you need \$30 for?	tóxico	neutro
women already have the right to vote, what more do they need?	tóxico	neutro

score superior em praticamente todos os casos, com exceção do Llama3 e Mistral utilizando o *prompt zero-shot*, onde os *F1-scores* obtidos foram levemente inferiores àquele obtido pelo HateBERT. Em relação ao RoBERTa, todos os LLMs atingiram um *F1-score* consideravelmente maior utilizando ambas as estratégias de *prompt*.

Dos resultados obtidos, é razoável concluir que os LLMs podem ser utilizados para a classificação de discurso de ódio implícito de forma satisfatória, pois, apenas utilizando engenharia de *prompt*, obtiveram desempenho equivalente ou superior aos modelos de arquitetura mais simples ajustados especificamente para a execução dessa tarefa.

É interessante notar também que, a utilização do *prompt few-shot* com os exemplos selecionados através de busca por similaridade no espaço de *embeddings*, gerou resultados positivos, aumentando o *F1-score* obtido pelos LLMs em todos os testes realizados.

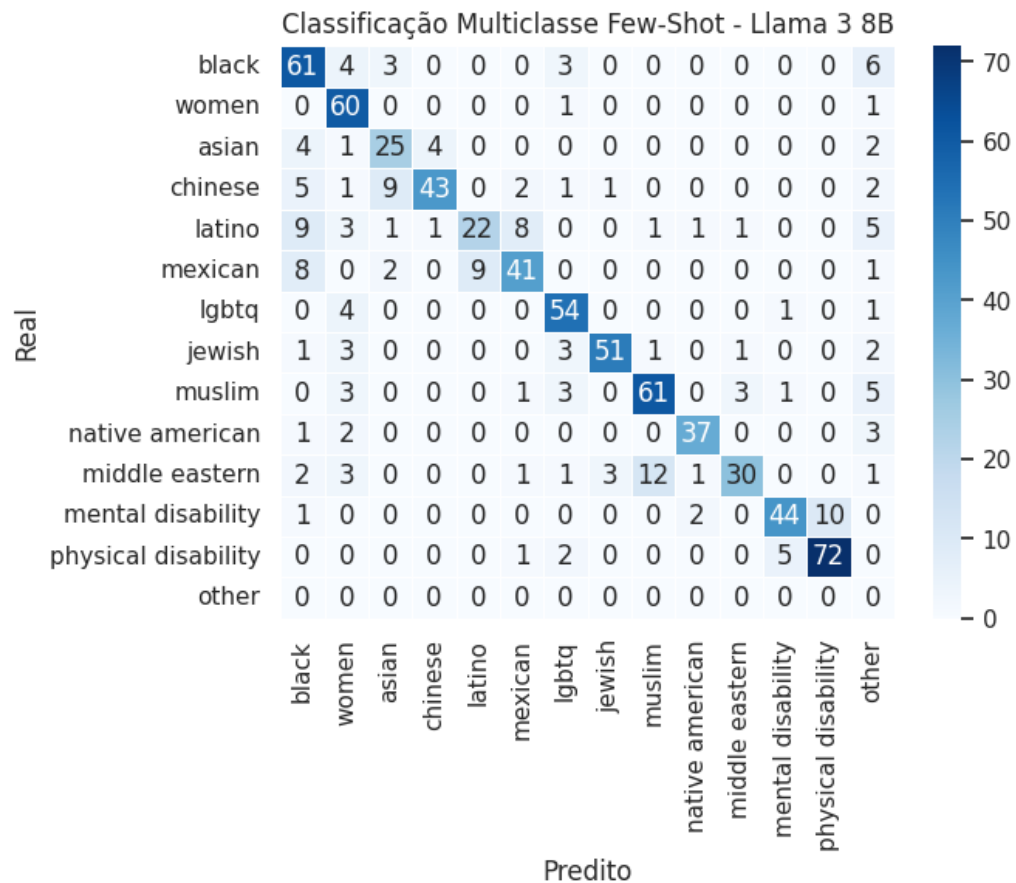
4.2 Classificação Multiclasse

Na tabela 4 estão apresentados os resultados da classificação multiclasse considerando todos os 13 grupos apresentados na seção 3.2.1, utilizando os *prompts zero-shot* e *few-shot*. A figura 33 mostra a matriz de confusão para o modelo Llama3 com o *prompt few-shot*, combinação que atingiu as melhores métricas. Tratando-se de LLMs, existe a possibilidade de os modelos gerarem como resposta uma classe diferente daquelas apresentadas a eles ou simplesmente não responderem com classe alguma. Por esse motivo e para considerar esses casos, foi criada a categoria *other*, que engloba todas as amostras às quais foram atribuídas categorias inválidas ou que o modelo não conseguiu classificar.

Observando a matriz de confusão, percebe-se que o modelo teve especial dificuldade em distinguir dentre alguns pares de grupo alvo – *asian* e *chinese*, *mexican* e *latino*, *muslim* e *middle-eastern*, *physical disability* e *mental disability* – confirmando as suspeitas

Tabela 4 – Métricas obtidas com os LLMs na classificação do grupo alvo

Prompt	Modelo	Acurácia	Precisão	Revocação	F1-Score
Zero-Shot	Llama3 8B	0,636	0,673	0,58	0,61
	Mistral 7B	0,566	0,654	0,526	0,54
	OpenChat 7B	0,659	0,642	0,601	0,609
Few-Shot	Llama3 8B	0,766	0,743	0,702	0,714
	Mistral 7B	0,711	0,727	0,655	0,679
	OpenChat 7B	0,743	0,71	0,685	0,691

Figura 33 – Matriz de confusão da classificação multiclasse utilizando o modelo Llama3 e *prompt few-shot*

levantadas na seção 3.2.1. Alguns exemplos dessas confusões podem ser conferidos na tabela 5. Examinando-os, é razoável afirmar que mais de uma categoria possa ser corretamente associada a cada exemplo, a depender da interpretação do texto, o que dificulta a classificação até mesmo por parte de seres humanos.

Para eliminar essa complexidade, que é desnecessária dados os objetivos deste projeto, foi realizada a agregação desses pares de classes semelhantes, conforme descrito na seção 3.2.1. Os resultados da classificação após a agregação das classes podem ser observados na tabela 6 e na figura 34, que mostra a matriz de confusão do modelo Llama3 com *prompt few-shot*, combinação com a qual os melhores resultados foram obtidos.

Tabela 5 – Confusões entre grupos alvo similares

Texto	Classe	Classe Predita
Go back to China.	asian	chinese
i dont want to see any more asians in this school	chinese	asian
Did you hear about the two car pile up in Mexico? 200 Mexicans died.	latino	mexican
i dont care if they are here legally or not, they should all be sent back	mexican	latino
america is a christian country, we should not allow muslims to come here	middle eastern	muslim
The congress support the Iran deal and it is a big step forward	muslim	middle eastern
if you have a child with a disability, it is best to put it up for adoption	mental disability	physical disability
People with disabilities are violent	physical disability	mental disability

Tabela 6 – Métricas obtidas com os LLMs na classificação dos grupos alvo agregados

Prompt	Modelo	Acurácia	Precisão	Revocação	F1-Score
Zero-Shot	Llama3 8B	0,724	0,73	0,67	0,682
	Mistral 7B	0,724	0,725	0,650	0,673
	OpenChat 7B	0,775	0,739	0,694	0,703
Few-Shot	Llama3 8B	0,847	0,772	0,767	0,763
	Mistral 7B	0,808	0,768	0,726	0,743
	OpenChat 7B	0,827	0,758	0,744	0,743

Nota-se que a utilização do *prompt few-shot* com exemplos selecionados dinamicamente novamente resultou no aumento de todas as métricas, independentemente do modelo e da agregação de classes. Nota-se também que a agregação das classes resultou em uma melhoria expressiva nos resultados, com o modelo Llama3 8B obtendo os melhores números em todas as métricas e um *F1-score* médio de 76,3%, considerado aceitável dada a quantidade de classes presentes e à dificuldade do conjunto de dados utilizado, que conta com amostras de texto geradas artificialmente e que são rotuladas de forma subjetiva, ainda que por avaliadores humanos.

Vale mencionar também que existem amostras no conjunto de dados que foram rotuladas incorretamente pelos avaliadores humanos, interferindo negativamente no cálculo das métricas mesmo que o modelo acerte as predições. Alguns exemplos desse tipo de amostra podem ser observados na tabela 7. Na tabela, a segunda coluna contém os grupos alvo ao quais os textos deveriam se referir e a terceira coluna contém as avaliações humanas referentes aos grupos mencionados no texto. Nota-se que em nenhum dos exemplos o texto gerado menciona o grupo alvo desejado.

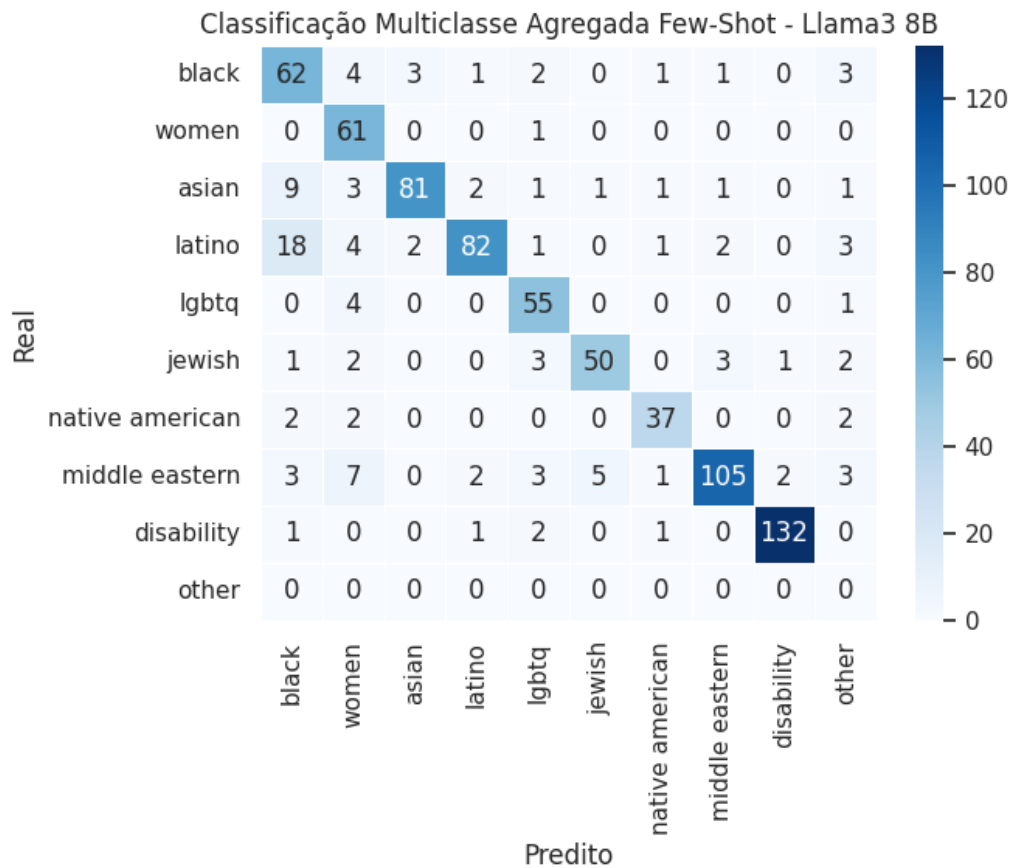


Figura 34 – Matriz de confusão da classificação multiclasse agregada utilizando o modelo Llama3 e *prompt few-shot*

Tabela 7 – Exemplos de amostras rotuladas incorretamente

Texto	Grupo Alvo	Grupo Alvo Predito
the iraq war is the most awesome thing ever	mexican	[‘the text references iraqi people’, ‘it is unclear who is referenced’, ‘it is unclear who is referenced’]
why should you never marry a jewish woman ? when dinner time comes , they ’re too scared to get near the oven	middle eastern	the text directly references middle eastern folks
there is no such thing as racism against white people, but there is such thing as racism against black people, but there is no such thing as racism	asian	[‘the text directly references asian folks’, ‘it is unclear who is referenced’, ‘the text directly references asian folks’]

4.3 Nível de Toxicidade

Para a avaliação dos níveis de toxicidade η , os níveis de toxicidade das amostras de texto foram discretizados segundo o critério indicado no diagrama da figura 9, conforme mencionado na seção 3.3.4, transformando o problema em uma classificação multiclasse. O

gráfico da figura 35 mostra a quantidade de amostras pertencentes a cada categoria após a discretização.

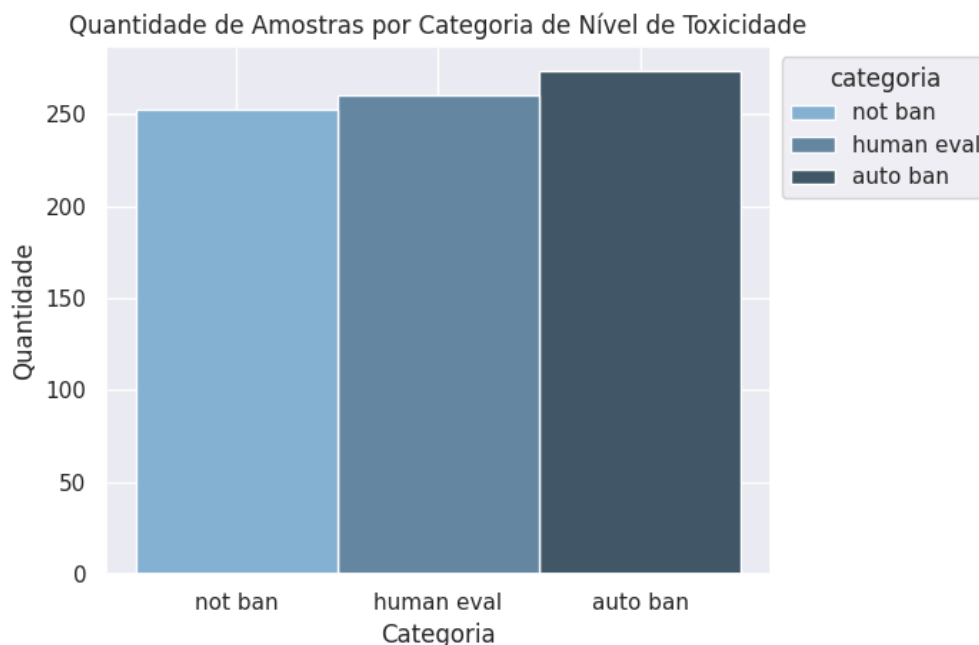


Figura 35 – Quantidade de amostras por categoria de nível de toxicidade

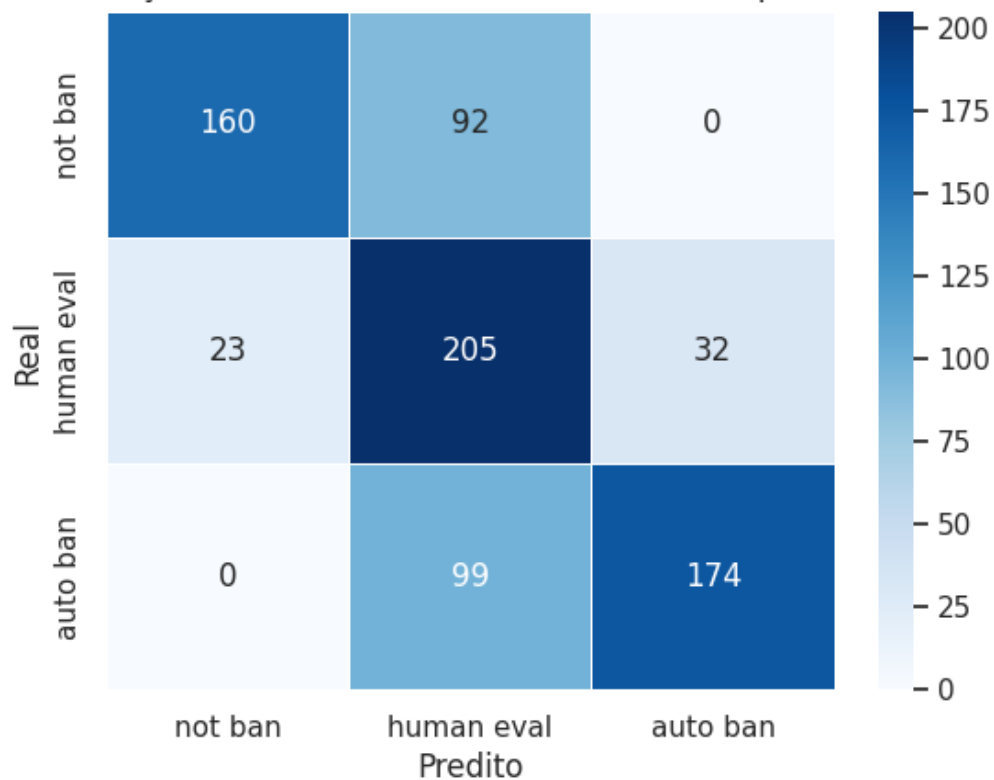
Em seguida, os níveis de toxicidade gerados pelos LLMs foram discretizados da mesma forma e as médias macro das métricas de acurácia, precisão, revocação e *F1-score* foram calculadas para cada um deles. Os resultados podem ser observados na tabela 8. O modelo Llama3, devido à censura, não conseguiu gerar respostas válidas para 4 das 785 amostras com o *prompt zero-shot*, e para 1 das 785 amostras com o *prompt few-shot*, o que corresponde a 0,51% e 0,13% do total de amostras presentes no conjunto de dados, respectivamente. Essas amostras foram ignoradas no cálculo das métricas do modelo.

Apesar de as melhores métricas terem sido obtidas utilizando o modelo Mistral 7B com o *prompt zero-shot*, por meio da análise das matrizes de confusão e dos tipos de erros cometidos pelos modelos, concluiu-se que a melhor combinação obtida foi o modelo OpenChat com o *prompt zero-shot*, cuja matriz de confusão pode ser observada na figura 36. A partir da matriz de confusão, pode-se observar que, apesar das métricas relativamente baixas, o modelo consegue diferenciar muito bem os textos muito tóxicos dos textos neutros, de forma que nenhuma confusão ocorreu entre eles. Isso é muito importante, pois as confusões entre esses tipos de texto são os erros mais graves possíveis que os modelos podem cometer. Também observa-se que a grande maioria dos erros cometidos pelo modelo foram na classificação incorreta de textos muito tóxicos ou neutros como textos que devem ser enviados para revisão, o tipo de erro menos grave de todos, por envolver intervenção humana no processo.

Tabela 8 – Métricas obtidas com os LLMs na classificação do nível de toxicidade

Prompt	Modelo	Acurácia	Precisão	Revocação	F1-Score
Zero-Shot	Llama3 8B	0,672	0,67	0,667	0,645
	Mistral 7B	0,731	0,725	0,729	0,712
	OpenChat 7B	0,687	0,746	0,687	0,696
Few-Shot	Llama3 8B	0,661	0,633	0,658	0,619
	Mistral 7B	0,679	0,657	0,677	0,648
	OpenChat 7B	0,712	0,711	0,713	0,708

Classificação do Nível de Toxicidade Zero-Shot - OpenChat 3.5

Figura 36 – Matriz de confusão da classificação dos níveis de toxicidade utilizando o modelo OpenChat e *prompt zero-shot*

Portanto, dados os tipos de erros cometidos e levando em consideração o auxílio de intervenção humana na maioria deles, é razoável concluir que o desempenho obtido pelo modelo na avaliação do nível de toxicidade dos textos é aceitável.

4.4 Explicações

A avaliação das explicações geradas, conforme mencionado na seção 3.3.5, foi feita de forma qualitativa. Como esperado, todos os modelos tiveram facilidade em explicar os motivos de os textos serem tóxicos ou não. Exemplos de explicações geradas pelos modelos para um texto tóxico e um texto não tóxico podem ser observadas nas tabelas 9 e 10, respectivamente. Na tabela 9, as explicações se referem ao texto “*women already have*

the right to vote, what more do they need?” e, na tabela 10, se referem ao texto “*women should be able to choose to have an abortion if they wish to do so*”.

Tabela 9 – Explicações geradas para um texto tóxico

Modelo	Explicação
OpenChat 7B	This text can be considered toxic because it assumes that women’s needs and rights are limited or less important than men’s. It disregards the ongoing struggle for gender equality, and undermines the importance of addressing various issues faced by women in society.
Mistral 8B	This text implies that women’s rights are already fully met, which is a harmful oversimplification of gender inequality issues. It dismisses ongoing struggles and the need for continued progress in women’s rights. Therefore, it is toxic towards the target group (women).
Llama3 8B	This text is toxic because it minimizes the struggles and ongoing efforts of women’s rights movements by implying that their most fundamental right has been fully realized. The comment also implies that women have already achieved equality, which is not only inaccurate but also dismissive of the continued gender-based injustices faced by many women.

Tabela 10 – Explicações geradas para um texto não tóxico

Modelo	Explicação
OpenChat 7B	The text is considered non-toxic because it promotes a personal choice and autonomy for women, which respects their rights and well-being. The statement does not target or insult any specific social group or individual.
Mistral 7B	This statement is non-toxic as it expresses a viewpoint about reproductive rights, which is a common topic of discussion. It does not contain offensive language or disrespect towards any individual or group.
Llama3 8B	This text is non-toxic because it promotes a socially accepted and legally protected right, ensuring autonomy over one’s own body. The statement is factual and free from insults or personal attacks. It also targets the specified social group, women, in a respectful manner.

É interessante notar que o texto “*women already have the right to vote, what more do they need?”* é o mesmo apresentado na tabela 3, classificado incorretamente pelo modelo OpenChat na etapa de classificação binária. Porém, o mesmo modelo foi capaz de explicar muito bem os motivos de o texto ser considerado tóxico, tendo as informações da categoria e grupo alvo fornecidas juntamente com o texto no *prompt*.

5 CONCLUSÕES

Após os experimentos realizados, é razoável concluir que todos os modelos escolhidos para a avaliação – Llama3 8B, Mistral 7B e OpenChat 7B – se mostraram capazes de diferenciar textos tóxicos de textos não tóxicos, superando modelos baseados na arquitetura BERT treinados especificamente para esse tipo de tarefa, como o HateBERT e RoBERTa, apenas utilizando estratégias de engenharia de *prompt*.

Adicionalmente, os mesmos modelos foram avaliados na identificação do grupo alvo mencionado nos textos, na avaliação do nível de toxicidade dos textos, e na geração de explicações a respeito dos textos, tarefas mais complexas do que uma simples classificação binária, e novamente se mostraram capazes de atingir um desempenho aceitável somente com engenharia de *prompt*, sem a necessidade de nenhum treinamento adicional.

Assim, pode-se afirmar que o objetivo deste projeto – o estudo da viabilidade da utilização de LLMs abertos e pequenos na identificação e avaliação de linguagem ofensiva e discurso de ódio implícito – foi atingido, e que o uso de LLMs para essa finalidade é viável, afinal.

5.1 Limitações

Devido a limitações de recursos computacionais e tempo, algumas decisões foram necessárias para tornar o desenvolvimento do projeto viável. Uma delas é a utilização de um único conjunto de dados para a avaliação dos modelos e do pipeline proposto. Embora os resultados dos experimentos sejam promissores, utilizar mais de um conjunto de dados é importante para avaliar os modelos em uma amostra de dados mais variada, o que permite análises mais robustas e com maior garantia de generalização. Além disso, o conjunto de dados utilizado foi gerado artificialmente, o que pode comprometer a qualidade das amostras, ainda que estas sejam avaliadas por humanos.

Além disso, não foi possível realizar uma análise extensiva na escolha dos *prompts* e dos parâmetros dos modelos utilizados nos experimentos. Os parâmetros padrão de todos os modelos foram mantidos e um único *prompt zero-shot* ou *few-shot* foi utilizado em cada experimento, para todos os modelos. A utilização de mais de um *prompt* ou combinação de parâmetros por modelo resultaria em uma quantidade proibitiva de experimentos e análises, dado o tempo disponível para o desenvolvimento do projeto. É evidente, no entanto, que todos os *prompts* foram elaborados cuidadosamente para cada experimento, ainda que definidos empiricamente.

Também devido a limitações de tempo e recursos computacionais, não foi possível realizar experimentos envolvendo o *fine-tuning* dos modelos para a execução das tarefas

específicas em que eles foram avaliados.

5.2 Melhorias e Trabalhos Futuros

Primeiramente, mais experimentos poderiam ser conduzidos utilizando conjuntos de dados diferentes, possibilitando mais avaliações e, conseqüentemente, maior segurança do desempenho dos modelos na execução da tarefa proposta.

Além disso, seria interessante explorar e incorporar mais técnicas de engenharia de *prompt* nos experimentos, obtendo *prompts* otimizados individualmente para cada modelo e para cada etapa do pipeline.

Por fim, poderiam ser empregadas técnicas de *fine-tuning* de LLMs conforme descrito em (Minaee *et al.*, 2024), criando versões dos modelos especializadas na execução das tarefas específicas de cada etapa do pipeline, buscando obter o melhor desempenho possível em cada uma delas.

REFERÊNCIAS

- ADL. **Hate is No Game: Hate and Harassment in Online Games 2022**. 2022. Disponível em: <https://www.adl.org/resources/report/hate-no-game-hate-and-harassment-online-games-2022>.
- AW, K. L. *et al.* **Instruction-tuning Aligns LLMs to the Human Brain**. 2023. Disponível em: <https://arxiv.org/abs/2312.00575>.
- BROWN, T. B. *et al.* **Language Models are Few-Shot Learners**. 2020. Disponível em: <https://arxiv.org/abs/2005.14165>.
- CASELLI, T. *et al.* **HateBERT: Retraining BERT for Abusive Language Detection in English**. 2021. Disponível em: <https://arxiv.org/abs/2010.12472>.
- CHIU, K.-L.; COLLINS, A.; ALEXANDER, R. **Detecting Hate Speech with GPT-3**. 2022. Disponível em: <https://arxiv.org/abs/2103.12407>.
- DAVIDSON, T. *et al.* **Automated Hate Speech Detection and the Problem of Offensive Language**. 2017. Disponível em: <https://arxiv.org/abs/1703.04009>.
- DEVLIN, J. *et al.* **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**. 2019. Disponível em: <https://arxiv.org/abs/1810.04805>.
- GUO, K. *et al.* **An Investigation of Large Language Models for Real-World Hate Speech Detection**. 2024. Disponível em: <https://arxiv.org/abs/2401.03346>.
- HARTVIGSEN, T. *et al.* **ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection**. 2022. Disponível em: <https://arxiv.org/abs/2203.09509>.
- HOWARTH, J. **How Many Gamers Are There? (New 2024 Statistics)**. 2024. Disponível em: <https://explodingtopics.com/blog/number-of-gamers>.
- JIANG, A. Q. *et al.* **Mistral 7B**. 2023. Disponível em: <https://arxiv.org/abs/2310.06825>.
- LEWIS, P. *et al.* **Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks**. 2021. Disponível em: <https://arxiv.org/abs/2005.11401>.
- LI, L. *et al.* “hot” chatgpt: The promise of chatgpt in detecting and discriminating hateful, offensive, and toxic comments on social media. **ACM Transactions on the Web**, Association for Computing Machinery (ACM), v. 18, n. 2, p. 1–36, mar. 2024. ISSN 1559-114X. Disponível em: <http://dx.doi.org/10.1145/3643829>.
- META. **Introducing Meta Llama 3: The most capable openly available LLM to date**. 2024. Disponível em: <https://ai.meta.com/blog/meta-llama-3>.
- MINAEE, S. *et al.* **Large Language Models: A Survey**. 2024. Disponível em: <https://arxiv.org/abs/2402.06196>.
- NG, L. H. X.; LIM, A. X. W.; YODER, M. M. **Challenges for Real-Time Toxicity Detection in Online Games**. 2024. Disponível em: <https://arxiv.org/abs/2407.04383>.

REIMERS, N.; GUREVYCH, I. **Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks**. 2019. Disponível em: <https://arxiv.org/abs/1908.10084>.

VASWANI, A. *et al.* **Attention Is All You Need**. 2023. Disponível em: <https://arxiv.org/abs/1706.03762>.

WANG, G. *et al.* **OpenChat: Advancing Open-source Language Models with Mixed-Quality Data**. 2024. Disponível em: <https://arxiv.org/abs/2309.11235>.

ZHAO, W. X. *et al.* **A Survey of Large Language Models**. 2023. Disponível em: <https://arxiv.org/abs/2303.18223>.